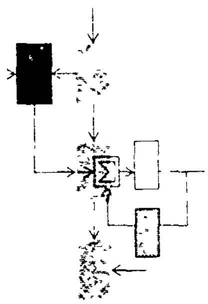
June, 1980

LIDS-R-1003

Research Supported By:
ONR Contract N00014-77-C-0532
OSP No. 85552



TARGET TRACKING BASED ON BEARING ONLY MEASUREMENTS

Svein Fagerlund

Laboratory for Information and Decision Systems
MASSACHUSETTS INSTITUTE OF TECHNOLOGY, CAMBRIDGE, MASSACHUSETTS 02139

11 June, 1980

14 LIDS-R-1003

12 236

16 TARGET TRACKING
BASED ON
BEARING ONLY MEASUREMENTS

by

10 Svein Fagerlund
A/S Kongsberg Våpenfabrikk,
Department F44
P. Box 25,
N-3601 Kongsberg,
NORWAY

and

Visiting Research Scientist
Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, MA. 02139

15 N00014-77-C-0532

Massachusetts Institute of Technology
Laboratory for Information and Decision Systems
Cambridge, MA 02139

410950

ACKNOWLEDGMENTS

This report represents the results of a six-month (part-time) effort to apply different approximative nonlinear filtering techniques on target tracking problem based on bearing only measurements.

The work was performed during the Academical Year 1979-80, when the author enjoyed his sabbatical year as Visiting Research Scientist at the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, Mass.

The author's stay at MIT was sponsored by A/S Kongsberg Våpenfabrikk, Kongsberg, Norway, and the Royal Norwegian Council for Scientific and Industrial Research, Oslo, Norway.

The author wishes to thank Asst. Prof. Robert R. Tenney, LIDS, for valuable input and discussions. Thanks also to LIDS' Director, Prof. Michael Athans, who admitted the author to LIDS, and was his overall supervisor during his stay at LIDS.

The research was supported in part by the Office of Naval Research under contract ONR/N00014-77-C-0532 (MIT OSP No. 85552).

ABSTRACT

This report develops 10 different approaches to the target tracking problem based on bearing only observations. Its purpose is to form the basis for an extensive simulation study, aimed at achieving the best possible tracking performance for this tracking problem. Included in the report are also a proposed initialization routine for bearing only trackers, and a maneuver detection algorithm with a proposed action scheme following the maneuver detection.

SUMMARY AND CONCLUSIONS

Ten different approximative nonlinear filtering approaches to the target tracking problem based on bearing only measurements have been developed.

These approaches include two Extended Kalman filters (Cartesian and Polar coordinate system representation), two second order Gaussian filters (Cartesian and Polar), five different iterated Extended Kalman filters, and one approach consisting of M parallel, Extended (Cartesian) Kalman filters.

The polar coordinate system representation of the target motion provided the necessary insight and equations for us to attack the initialization problem based on bearing only information. Based on the two assumptions: 1) Straight target trajectory and 2) Noisefree bearings, we could show that given a target range, the target velocity components could be calculated from three consecutive bearings, with no constraints on the observer's trajectory over the observation interval. We could also show, that if the observer's velocity was zero, we could calculate the target's heading exact, even if the assumed range was false.

By including a fourth bearing, and introducing the constraint that the observer should be maneuvering during the observation interval, we also derived the equation for calculation of range to target.

In order to take into account the fact that the observations are noisy, a procedure for smoothing of initial range and velocity data calculated from a batch of bearing observations was proposed.

In order to adapt the straight trajectory filters to the curved trajectory case, which results from a maneuvering target, we derived the structure for a likelihood ratio test based on the principle of comparing the innovation sequence with its expected statistics, based on the "no maneuvre" hypothesis. When the actual variance of the innovation sequence becomes consistently larger than its expected variance over the most recent L samples, the "target maneuvre" hypothesis is accepted.

The proposed action scheme following the maneuvre detection included two main features:

1. A reprocessing of the observations obtained in the time period Δt prior to the maneuver detection, where Δt is the nominal time delay between the start of the maneuvre and its detection.
2. Imposing limited memory on the filter, by increasing the velocity elements of the covariance matrix, before the reprocessing of observations described above takes place.

The result of this scheme will be a better utilization of the observations obtained during the target maneuvre. When the reprocessing of the observations obtained during the time period Δt are finished, a discrete jump to a more correct position and velocity of the state vector at the time when the maneuvre was detected, will be the result.

The purpose of this report is to be a mathematical basis for an extensive simulation study where the performance of the different filtering approaches to bearing only tracking can be compared.

When this simulation study has been performed, we will be in the position to give an answer to the question: To what extent is it possible to overcome the three main problems with bearing only tracking: 1) The poor degree of observability, 2) The nonlinearity of the problem, claiming for the best possible linearization trajectory, and 3) The target maneuvre detection and handling problem.

We have not made any effort to analyze and compare the computational complexity nor the memory capacity requirements of the different filters. The reason for this is that we didn't want these aspects of the problem to constraint our choice of filtering approaches at this stage.

CONTENTS

TABLE OF CONTENTS (cont'd)

TABLE OF CONTENTS (cont'd)

1. INTRODUCTION

1.1 Problem Statement

Our objective is to investigate the target tracking problem based on bearing observations from a single, moving observer.

Our approach will be threefold:

1. Derive different mathematical approaches to the filtering problem for targets moving along straight trajectories,
2. Derive more optimal initialization routines based only on bearing observations from the moving observer.
3. Derive maneuvre detection algorithms, which, together with specific action patterns following the maneuvre detection, allows the straight trajectory filters to adapt to curved trajectories.

We restrict the problem to tracking in the x-y-plane.

The observer's position, velocity and acceleration are known functions of time, with known accuracy (1σ values).

The bearing sensor's accuracy is known (1σ).

1.2 Report Outline

The composition of this report can be described as follows:

Chapter 2 introduces the difficulties that exists for this type of tracking, and gives an outline of the different approaches to straight trajectory tracking that will be developed.

In Chapter 3 we develop the mathematical equations for 10 different filtering approaches to this tracking problem.

In Chapter 4 the initialization problem is analyzed, and necessary equations for an initialization routine for bearing only trackers are proposed.

Chapter 5 gives a short resume over earlier approaches to maneuvering target tracking, and continues with the detailed equations for a proposed maneuvre detection and handling scheme for the different filtering approaches given in Chapter 3.

The purpose of this report is to form the basis for a simulation study, where the different tracking approaches are compared, and the filtering approach with the best overall performance can be selected. Chapter 6 gives a few guidelines to be taken into account while planning and performing this simulation study.

2. PROBLEM ANALYSIS AND APPROACH PROPOSALS

The main problem with this type of tracking is the poor degree of observability. It is well known that the success of this tracking scheme depends entirely on the observer's maneuvering. Simultaneously, in most cases, the tracker has to be based on a maneuvering scheme selected on other criterias than tracking performance. This is due to reasons like:

- Narrow waters
- Not reveal the observer's position
- Not restrict the captain's decision space.

The topic of selecting the optimal maneuvre in order to maximize the observability has, however, been treated by D.J. Murphy [1].

Another problem is the necessity to perform linearization about the target trajectory. However, this trajectory is unknown. The purpose of our tracker is to estimate this trajectory. Now, if linearization is performed about the a priori estimate of the state vector, which is the most obvious thing to do, the utilization of observations, when the estimated trajectory is far from the correct one, will be poor. This is the case in the initialization phase, and after a target maneuvre.

A proposed initialization routine for bearing only trackers will be developed in Chapter 4.

The third main problem with this type of tracking is the maneuvre detection and handling problem. If the target performs a maneuvre after a stable target solution is established, it is possible to detect the occurrence of the maneuvre (after a certain period of time) without the observer maneuvering. However, in order to arrive at the new, post-maneuvre target course and speed, an observer maneuvre normally has to take place. This is due to the time delay between the maneuvre occurrence and its detection, resulting in range to target error at the target maneuvre detection time point. Since this time delay will vary, depending on the type and size of the target maneuvre, the range/velocity ambigu. v cannot be resolved exactly without an observer maneuvre. Maneuvre detection and following actions patterns will be proposed in Chapter 5.

The question is now: How can these three problems be overcome? Or to put it more correctly: To what extent do different approaches to this target tracking scheme overcome these three problems?

Existing literature on the subject fail to give an answer to this question.

Our intention is, therefore, to develop a number of different approaches to this target tracking problem, which can form the basis for an extensive simulation study, where the different approaches are compared. When this simulation study is finished, our basis for giving a reliable answer to the question above should be greatly improved.

The following 10 approaches are proposed:

1. Extended Kalman filter, Cartesian Coordinate system representation. State vector:

$$\underline{x}^c = \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix} \tag{2.1}$$

observation equation:

$$\phi = \tan^{-1}\left(\frac{x - x_s}{y - y_s}\right) + w \qquad (2.2)$$

where

x = target's x - coordinate

y = target's y - coordinate

v_x = target's x - velocity

v_y = target's y - velocity

x_s = observer's x - coordinate

y_s = observer's Y - coordinate

ϕ = bearing from observer to target rel. north.

w = observation noise

2. Extended Kalman filter, polar coordinate system representation.

State vector:

$$\underline{x}^P = \begin{bmatrix} \phi \\ R \\ v_x \\ v_y \end{bmatrix} \qquad (2.3)$$

Observation equation:

$$\phi = [1 \quad 0 \quad 0 \quad 0]\underline{x}^P + w \qquad (2.4)$$

where ϕ, v_x, v_y and w are defined earlier, and

R = range from observer to target

3. Second order Gaussian filter, representation as in 1.

4. Second order Gaussian filter, representation as in 2.

5. Iterated Extended Kalman filter, representation as in 1, iteration scheme as in Jazwinski [2], pp. 278-279.

6. Iterated Linear Filter-Smoother. Representation as in 2, iteration scheme as in Jazwinski [2], pp. 279-281.

7. Global Iterated Filter, Representation as in 1, iteration scheme as proposed by Jazwinski [2], pp. 281.

8. Global Iterated Filter. Representation as in 2, iteration scheme as in 7.

9. Extended Kalman filter, Cartesian coordinate system representation. State vector:

$$\underline{\tilde{x}}_k = \begin{bmatrix} \underline{x}_k^c \\ \hline \underline{x}_{k-N}^c \end{bmatrix} = \begin{bmatrix} x_k \\ y_k \\ v_{x_k} \\ v_{y_k} \\ x_{k-N} \\ y_{k-N} \\ v_{x_{k-N}} \\ v_{y_{k-N}} \end{bmatrix} \tag{2.5}$$

where

N = May be variable number ≥ 1

$\underline{x}_k^c$ = State vector at sample k, representation as in 1.

$\underline{x}_{k-N}^c$ = State vector at sample k-N, representation as in 1.

Observation equation:

$$\underline{z}_k = \begin{bmatrix} \phi_k \\ \phi_{k-N} \end{bmatrix} = \begin{bmatrix} \tan^{-1}\left(\dfrac{x_k - x_{sk}}{y_k - y_{sk}}\right) \\ \tan^{-1}\left(\dfrac{x_{k-N} - x_{sk-N}}{y_{k-N} - y_{sk-N}}\right) \end{bmatrix} + \begin{bmatrix} w_k \\ w_{k-N} \end{bmatrix} \tag{2.6}$$

This scheme will process each observation twice. Details outlined in Chapter 3.6.3 ("Serial" filter approach)

10. Parallel filter approach. M filters, each with Cartesian coordinate system representation as in approach 1, are initialized with range

$$R_i = R_o + i \cdot \Delta R, \quad i = 1,2,\ldots,M \tag{2.7}$$

where R_0 and ΔR are constants. The a priori variances on R_i for each filter are assumed low, in order to stablize range for each filter. The resulting state vector for this filtering scheme is given by:

$$\hat{\underline{x}}_k = \sum_{i=1}^{M} p(R_i = R/z_k)\hat{\underline{x}}_{ki} \tag{2.8}$$

where the probability $p(R_i = R/z_k)$ has to be determined. Details in Chapter 3.7.

3. SYSTEM EQUATIONS FOR THE DIFFERENT APPROACHES

We will now continue with an outline of the system and filter equations for the different approaches. However, special equations for tracker initialization and for maneuvre detection and handling will not be addressed in this chapter. These subjects will be treated in Chapter 4 and 5, respectively.

3.1 Extended Kalman filter, Cartesian Coordinate System.

The Cartesian Coordinate system representation is the most common way to model the target motion dynamics. Several papers on target tracking use this representation [3]-[11], and most probably in the majority of the existing target tracking systems of this kind in operating order, this representation is used.

Assuming the target is nonmaneuvering, the following mathematical model can be established:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{v}_x \\ \dot{v}_y \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ v_x \\ v_y \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \tag{3.1}$$

$$z = \phi = \tan^{-1}\left(\frac{x - x_s}{y - y_s}\right) + w \tag{3.2}$$

The geometrical situation is depicted in Fig. 3.1:

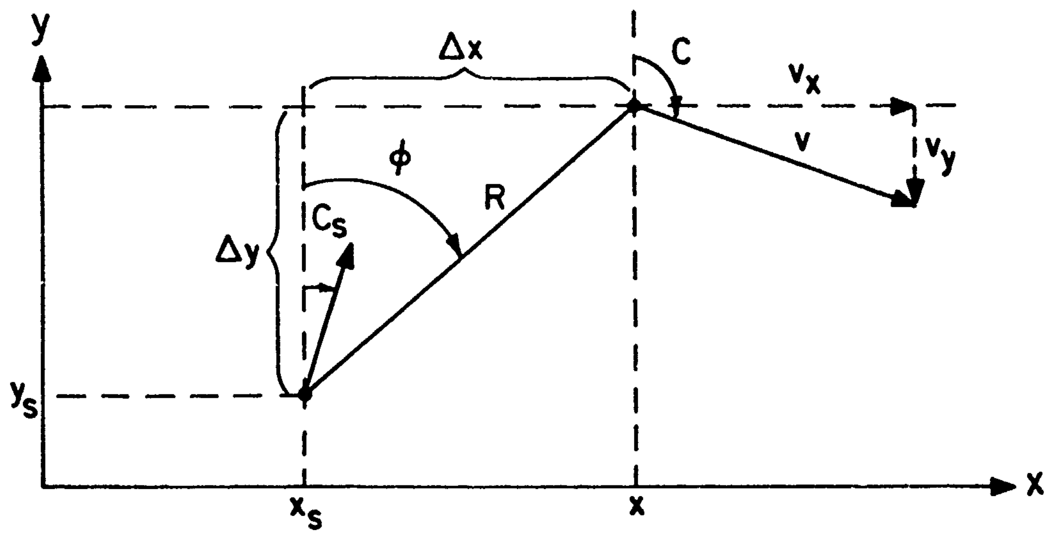


Fig. 3.7. Target-Observer-Geometry.

The different variables in equations (3.1) and (3.2) should be self-explaining from Fig. 3.1, except for w and $\underline{v} = [v_1\ v_2]^T$. These are the measurement and the process-noise, respectively.

Since the tracker is going to be realized on a digital computer, the discrete version of the equations (3.1) and (3.2) are sought. We get (1/T - sample frequency) [3]:

$$\underline{x}_{k+1} = \phi(T)\underline{x}_k + \theta(T)\underline{v}_k \tag{3.3}$$

$$z_k = g(\underline{x}_k) + w_k \tag{3.4}$$

Here, $\phi(T)$, $\theta(T)$ and $g(\underline{x}_k)$ are defined by:

$$\phi(t) = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.5}$$

$$\theta(T) \begin{bmatrix} T^2/2 & 0 \\ 0 & T^2/2 \\ T & 0 \\ 0 & T \end{bmatrix} \tag{3.6}$$

$$g(\underline{x}_k) = \tan^{-1}\left(\frac{\Delta x_k}{\Delta y_k}\right) \tag{3.7}$$

Further, $\underline{v}_k$ and w_k are assumed to be while Gaussian processes with the following characteristics:

$$\left.\begin{array}{l} E\{\underline{v}_k\} = \underline{0} \\ E\{w_k\} = 0 \\ E\{\underline{v}_k \underline{v}_j^T\} = V_k \ \sigma_{kj} \\ E\{w_k^2\} = W_k \\ \sigma_{kj} = \begin{Bmatrix} 1 & k=j \\ 0 & k\neq j \end{Bmatrix} \end{array}\right\} \quad \begin{array}{r} (3.7) \\ \\ (3.8) \end{array}$$

The noise processes $\underline{v}_k$ and w_k are assumed uncorrelated. The process noise covariance matrix, V_k, has the following form:

$$V_k = \begin{bmatrix} v_{1k} & 0 \\ 0 & v_{2k} \end{bmatrix} \tag{3.9}$$

Further, we define:

$$H_k = \left.\frac{\partial g}{\partial \underline{x}_k}\right|_{\underline{x}_k = \hat{\underline{x}}_{k,k-1}} = \begin{bmatrix} \dfrac{\Delta y_k}{R_k^2}, & \dfrac{-\Delta x_k}{R_k^2} & 0 & 0 \end{bmatrix} \tag{3.10}$$

where $\hat{\underline{x}}_{k,k-1}$ is the a priori estimate of the state vector at time k.

The Extended Kalman filter equations for the system described by equations (3.3) to (3.10) are the following:

Initialization:

$$\hat{\underline{x}}_{0,-1} = \underline{x}_0^c \tag{3.11}$$

$$P_{0,-1} = P_0^c \tag{3.12}$$

Observation integration:

$$\hat{\underline{x}}_{k,k} = \hat{\underline{x}}_{k,k-1} + K_k(z_k - \hat{z}_{k,k-1}) \tag{3.13}$$

$$\hat{z}_{k,k-1} = g(\hat{\underline{x}}_{k,k-1}) \tag{3.14}$$

$$K_k = P_{k,k-1}H_k^T(H_kP_{k,k-1}H_k^T + W_k)^{-1} \tag{3.15}$$

$$P_{k,k} = (I-K_kH_k)P_{k,k-1}(I-K_kH_k)^T + K_kW_kK_k^T \tag{3.16}$$

Time updating:

$$\hat{\underline{x}}_{k+1,k} = \phi(T)\hat{\underline{x}}_{k,k} \tag{3.17}$$

$$P_{k+1,k} = \phi(T)P_{k,k}\phi(T)^T + \theta(T)V_k\theta(T)^T \tag{3.18}$$

3.2 Extended Kalman filter, Polar Coordinate System.

A discrete Polar coordinate system representation of the non-maneuvering target can be derived mathematically, however, the easiest way is to use a geometrical approach. Fig. 3.2 shows the relative geometrical situation, as seen from the observer (the coordinate system is fixed to the observer):

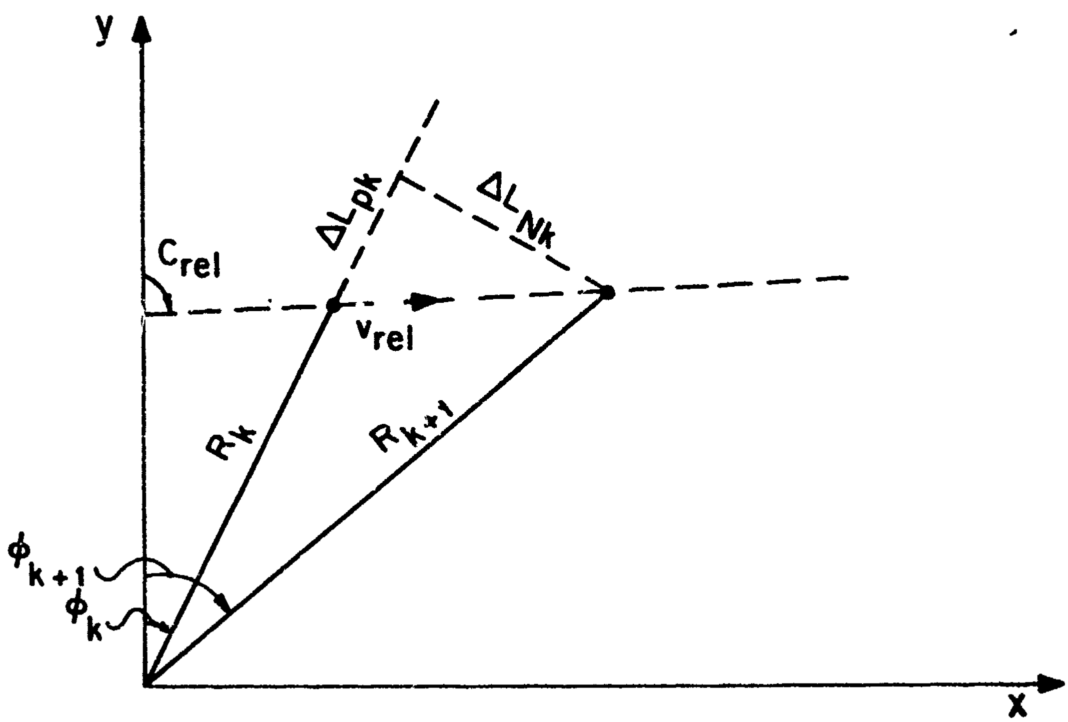


Fig. 3.2. Relative Geomettical Situation.

By refering to Fig. 3.2, the following two equations can be established directly:

$$\phi_{k+1} = \phi_k + \tan^{-1}\left(\frac{\Delta L_{Nk}}{R_k+\Delta L_{pk}}\right) + v_{1k} \tag{3.19}$$

$$R_{k+1} = \sqrt{(R_k+\Delta L_{pk})^2 + \Delta L_{Nk}^2} + v_{2k} \tag{3.20}$$

where v_{1k} and v_{2k} are additional white Gaussian noise processes.

The variables ΔL_{pk} and ΔL_{Nk} are the relative displacement between observer and target over the sample period [kT, (k+1)T], along and across the line of sight to target, ϕ_k. These variables can be given by:

$$\begin{bmatrix} \Delta L_{Nk} \\ \Delta L_{pk} \end{bmatrix} = S_k \cdot \left(\begin{bmatrix} \Delta x_{Tk} \\ \Delta y_{Tk} \end{bmatrix} - \begin{bmatrix} \Delta x_{sk} \\ \Delta y_{sk} \end{bmatrix} \right) \tag{3.21}$$

where the transformation matrix, s_k, is given by:

$$S_k = \begin{bmatrix} \cos \phi_k & -\sin \phi_k \\ \sin \phi_k & \cos \phi_k \end{bmatrix} \tag{3.22}$$

and $[\Delta x_{Tk} \Delta y_{Tk}]^T$ and $[\Delta x_{sk}\ \Delta y_{sk}]^T$ are the target's and the observer's absolute displacement in x,y - direction during the sample period [kT, (k+1)T].

Since the assumption is made that the target is not maneuvering, we have:

$$\begin{bmatrix} \Delta x_{Tk} \\ \Delta y_{Tk} \end{bmatrix} = T \cdot \begin{bmatrix} v_{xk} \\ v_{yk} \end{bmatrix} \tag{3.23}$$

where

v_{xk} = target absolute velocity in x-direction at time kT

v_{yk} = target absolute velocity in y-direction at time kT

The observer's displacement during the sample period $[kT, (k+1)T]$, $[\Delta x_{sk} \Delta y_{sk}]^T$, is given by the observer's dead reckoning system. Assuming that the observer's acceleration is constant over the sample period, we can write:

$$\begin{bmatrix} \Delta x_{sk} \\ \Delta y_{sk} \end{bmatrix} = \begin{bmatrix} v_{sxk} \\ v_{syk} \end{bmatrix} \cdot T + \frac{1}{2} T^2 \cdot \begin{bmatrix} a_{sxk} \\ a_{syk} \end{bmatrix} \tag{3.24}$$

where

v_{sxk} = observer's velocity in the x-direction at time kT

v_{syk} = observer's velocity in the y-direction at time kT

a_{syk} = observer's acceleration in the x-direction at time kT

a_{syk} = observer's acceleration in the y-direction at time kT

Now, selecting the Cartesian velocity components v_{xk} and v_{yk} as representation for the target velocity, the total, nonlinear Polar coordinate system representation of the target-observer relationship, is given by:

$$\underline{x}^p_{k+1} = \begin{bmatrix} \phi_{k+1} \\ R_{k+1} \\ v_{x\,k+1} \\ v_{y\,k+1} \end{bmatrix} = \begin{bmatrix} \phi_k + \tan^{-1}\left(\dfrac{\Delta L_{Nk}}{R_k + \Delta L_{pk}}\right) \\ \sqrt{(R_k + \Delta L_{pk})^2 + \Delta L_{Nk}^2} \\ v_{xk} \\ v_{yk} \end{bmatrix} + \begin{bmatrix} v_{1k} \\ v_{2k} \\ v_{3k} \\ v_{4k} \end{bmatrix} \tag{3.25}$$

Defining:

$$\underline{f}(\underline{x}_k^p) = \begin{bmatrix} \phi_k + \tan^{-1}\left(\dfrac{\Delta L_{Nk}}{R_k + \Delta L_{pk}}\right) \\ \sqrt{(R_k + L_{pk})^2 + \Delta L_{Nk}^2} \\ v_{xk} \\ v_{yk} \end{bmatrix} \tag{3.26}$$

and

$$\underline{v}_k = [v_{1k} v_{2k} v_{3k} v_{4k}]^T \tag{3.27}$$

equation (3.25) can be written:

$$\underline{x}_{k+1}^p = \underline{f}(\underline{x}_k^p) + \underline{v}_k \tag{3.28}$$

Further, defining the observation matrix:

$$H = [1 \quad 0 \quad 0 \quad 0] \tag{3.29}$$

the observation equation for the system described by equation (3.28) will be:

$$z_k = H \cdot \underline{x}_k^p + w_k \tag{3.30}$$

Equations (3.28) and (3.30) are our final equations for the Polar coordinate system version of the target tracking problem. As we can see, the system dynamics are nonlinear, while the observation equation is linear.

In order to utilize the Extended Kalman filter on this system, we have to develop:

$$F_k = \left.\frac{\partial \underline{f}(\underline{x}_k^p)}{\partial \underline{x}_k^p}\right|_{\underline{x}_k^p = \hat{\underline{x}}_{k,k}^p} \tag{3.31}$$

The development of F_k is done in APPENDIX A. The result is:

$$F_k = \begin{bmatrix} \frac{R_k(R_k + \Delta L_{pk})}{R_{k+1}^2} & \frac{\Delta L_{Nk}}{R_{k+1}^2} & \frac{T.\Delta L_{yk}}{R_{k+1}^2} & -\frac{T\Delta L_{xk}}{R_{k+1}^2} \\ \frac{R_k \cdot \Delta L_{Nk}}{R_{k+1}} & \frac{R_k + \Delta L_{pk}}{R_{k+1}} & \frac{T.\Delta L_{xk}}{R_{k+1}} & \frac{T\Delta L_{yk}}{R_{k+1}} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.32}$$

where

$$\Delta L_{xk} = (R_k + \Delta L_{pk}) \sin \phi_k + \Delta L_{Nk} \cdot \cos \phi_k \tag{3.33}$$

$$\Delta L_{yk} = (R_k + \Delta L_{pk}) \cos \phi_k - \Delta L_{Nk} \sin \phi_k \tag{3.34}$$

or, by inserting for ΔL_{pk} and ΔL_{Nk} from equation (3.21):

$$\Delta L_{xk} = R_k \cdot \sin \phi_k + \Delta x_{Tk} - \Delta x_{sk} \tag{3.35}$$

$$\Delta L_{yk} = R_k \cos \phi_k + \Delta y_{Tk} - \Delta y_{sk} \tag{3.36}$$

The Extended Kalman filter-equations for the system described by equations (3.28) and (3.30) are the following:

Initialization:

$$\hat{\underline{x}}^p_{0,-1} = \underline{x}^p_0 \tag{3.37}$$

$$P^p_{0,-1} = P^p_0 \tag{3.38}$$

Observation Integration:

$$\hat{\underline{x}}^p_{k,k} = \hat{\underline{x}}^p_{k,k-1} + K_k(z_k - \hat{z}_{k,k-1}) \tag{3.39}$$

$$\hat{z}_{k,k-1} = H\hat{\underline{x}}^p_{k,k-1} \tag{3.40}$$

$$K_k = P^p_{k,k-1}H^T(HP^p_{k,k-1}H^T + W_k)^{-1} \tag{3.41}$$

$$P^p_{k,k} = (I-K_kH)P^p_{k,k-1}(I-K_kH)^T + K_kW_kK^T_k \tag{3.42}$$

Time updating:

$$\hat{\underline{x}}^p_{k+1,k} = \underline{f}(\hat{\underline{x}}^p_{k,k}) \tag{3.43}$$

$$P^p_{k+1,k} = F_kP^p_{k,k}F^T_k + V^p_k \tag{3.44}$$

where the process noise covariance matrix, V^p_k, is given by

$$V^p_k = \begin{bmatrix} v^p_{1k} & 0 & 0 & 0 \\ 0 & v^p_{2k} & 0 & 0 \\ 0 & 0 & v^p_{3k} & 0 \\ 0 & 0 & 0 & v^p_{4k} \end{bmatrix} \tag{3.45}$$

In order to get identical initial conditions for the two representations given in Section 3.1 and 3.2, the following relationship exists between the initial conditions given in equations (3.11), and (3.37), (3.38):

$$\underline{x}_0^c = \begin{bmatrix} 0 & \sin\phi_0 & 0 & 0 \\ 0 & \cos\phi_0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \underline{x}_0^p \tag{3.46}$$

Further, defining:

$$T = \begin{bmatrix} R\cos\phi_0 & \sin\phi_0 & 0 & 0 \\ -R\sin\phi_0 & \cos\phi_0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.47}$$

we have:

$$P_0^c = TP_0^pT^T \tag{3.48}$$

3.3 Second Order Gaussian Filters.

The following development follows the spirit of Jazwinski [2], pp. 91, 336-346, 362-365, and Gelb [12], pp. 191-192. We are, however, interested in a system representation where both system dynamics and observation are discrete, while [2] and [12] are concerned with continuous system representation, and discrete observations. Therefore, the discrete version of the second order time updating equations have to be developed.

Jazwinski [2] defines 4 different possible solutions to second order filtering:

1. The truncated second order filter.

2. The Gaussian second order filter.

3. The modified truncated second order filter.

4. The modified Gaussian second order filter.

We select approach no. 4, which also has been developed by Wishner et al. [13]. For this approach the following assumptions are made about $p(x_k)$;

1. $p(x_k)$ is symmetric and "close to the mean".

2. The third central moment is zero.

3. Assuming Gaussian density, the fourth central moment is approximated by:

$$E\left\{(\underline{x}_i-\hat{\underline{x}}_i)(\underline{x}_j-\hat{\underline{x}}_j)(\underline{x}_k-\hat{\underline{x}}_k)(\underline{x}_\ell-\hat{\underline{x}}_\ell)\right\} = P_{jk}P_{i\ell} + P_{j\ell}P_{ik} + P_{k\ell}P_{ij} \tag{3.49}$$

4. The fifth and higher order moments are neglected.

In order to proceed the development, a couple of operators have to be defined, namely 1) $\partial_x^2(\underline{f}(\underline{x}),B)$ and 2) $(\partial^2\underline{f}\ P^2\partial^2\underline{f})$:

1) The operator $\partial_x^2(\underline{f}(x),B)$, for any function $\underline{f}(x)$, any $\underline{x}$ and any matrix B is a vector whose i^{th} element is defined by:

$$\partial_{xi}^2(\underline{f},B) \triangleq \text{trace}\left\{\frac{\partial}{\partial\underline{x}}\left[\frac{\partial\underline{f}_i}{\partial\underline{x}}\right]^T \cdot B\right\} \tag{3.50}$$

2) The operator $[\partial_{\underline{x}}^2\underline{f}(\underline{x})P^2\partial_x^2 f(\underline{x})]$, for any function $f(\underline{x})$, any $\underline{x}$ and any symmetric matrix P, is a symmetric matrix with elements (i,j) [2]:

$$\sum_{k,\ ,p,q}^{n} \frac{\partial^2 f_i}{\partial x_k \partial x_\ell} P_{\ell p} P_{kq} \frac{\partial^2 f_j}{\partial x_p \partial x_q} \tag{3.51}$$

Wishner et al [13] gives another definition of this operator, which turn out to give identical result. The $[i,j]^{th}$ element is in [13] given by:

$$\text{trace}\left\{\frac{\partial}{\partial \underline{x}}\left[\frac{\partial}{\partial \underline{x}}\left(\underline{f}_i(\underline{x})\right)\right]^T \cdot P \frac{\partial}{\partial \underline{x}}\left[\frac{\partial}{\partial \underline{x}}\left(f_j(\underline{x})\right)\right]^T \cdot P\right\} \tag{3.52}$$

In the following, we will use the definition in eq. (3.52) for this operator.

With the definition of these two operators in mind, we are ready to develop the modified Gaussian second order filter.

The system and observation equations are assumed to have the following form:

$$\underline{x}_{k+1} = \underline{f}(\underline{x}_k) + \underline{v}_k \tag{3.53}$$

$$z_k = g(\underline{x}_k) + w_k \tag{3.54}$$

If $\underline{f}$ and g are sufficiently smooth, the second order terms can be included in the Taylor expansions for $\underline{f}$ and g. Linearizing $\underline{f}$ about $\underline{x}_k = \hat{\underline{x}}_{k,k}$ and g about $\underline{x}_k = \hat{x}_{k,k-1}$, we get:

$$\begin{aligned}\underline{f}(\underline{x}_k) &= \underline{f}(\hat{\underline{x}}_{k,k}) + \left.\frac{\partial \underline{f}}{\partial \underline{x}}\right|_{x = \hat{x}_{k,k}} (\underline{x}_k - \hat{\underline{x}}_{k,k}) \\ &+ \frac{1}{2}\left.\partial^2_{\underline{x}}\right|_{\underline{x} = \hat{\underline{x}}_{k,k}} (\underline{f}, (\underline{x}_k - \hat{\underline{x}}_{k,k})(\underline{x}_k - \hat{\underline{x}}_{k,k})^T)\end{aligned} \tag{3.55}$$

$$\begin{aligned}g(x_k) &= g(\hat{\underline{x}}_{k,k-1}) + \left.\frac{\partial g}{\partial \underline{x}}\right|_{\underline{x} = \hat{\underline{x}}_{k,k-1}} (\underline{x}_k - \hat{\underline{x}}_{k,k-1}) \\ &+ \frac{1}{2}\left.\partial^2_{\underline{x}}\right|_{\underline{x} = \hat{\underline{x}}_{k,k-1}} (g, (\underline{x}_k - \hat{\underline{x}}_{k,k-1})(x_k - \hat{\underline{x}}_{k,k-1})^T)\end{aligned} \tag{3.56}$$

We recall the definitions of equations (3.10) and (3.31). Now, taking the expectations of equations (3.55) and (3.56), making use of the assumptions that:

$$E\{\underline{x}_k - \hat{\underline{x}}_{k,k}\} = \underline{0}$$
$$E\{\underline{x}_k - \hat{\underline{x}}_{k,k-1}\} = \underline{0} \tag{3.57}$$

we get:

$$\hat{\underline{x}}_{k+1,k} = E\{\underline{f}(\underline{x}_k)\} = \underline{f}(\hat{\underline{x}}_{k,k}) + \frac{1}{2}\,\partial_{\hat{\underline{x}}_{k,k}}(\underline{f},\cdot P_{k,k}) \tag{3.58}$$

$$\hat{z}_{k,k-1} = E\{g(x_k)\} = g(\hat{\underline{x}}_{k,k-1}) + \frac{1}{2}\,\partial_{\hat{\underline{x}}_{k,k-1}}(g,P_{k,k-1}) \tag{3.59}$$

Equation (3.58) is the second order approximation difference equation for $\underline{x}_k$ between observations. We now seek a recursive equation for the covariance matrix P_k between observations. Using (3.53), (3.55), (3.31) and (3.58) we get:

$$P_{k+1,k} = E\{(\underline{x}_{k+1}-\hat{\underline{x}}_{k+1,k})(\underline{x}_{k+1}-\hat{\underline{x}}_{k+1,k})^T\}$$
$$= E\{[F_k\cdot\Delta\hat{\underline{x}}_{k,k} + \frac{1}{2}\,\partial^2_{\hat{\underline{x}}_{k,k}}(\underline{f},\ \Delta\hat{\underline{x}}_{k,k}\cdot\Delta\hat{x}^T_{k,k}) - \frac{1}{2}\,\partial^2_{\hat{\underline{x}}_{k,k}}(\underline{f},P_{k,k}) + \underline{v}_k]\cdot[\cdot]^T\}$$
$$= F_kP_{k,k}F_k^T + V_k + L_k \tag{3.60}$$

where L_k is given by:

$$L_k = \frac{1}{4}\,E\{\partial^2_{\hat{\underline{x}}_{k,k}}(\underline{f},\ \Delta\hat{\underline{x}}_{k,k}\Delta\hat{\underline{x}}^T_{k,k})\cdot\partial^2_{\hat{\underline{x}}_{k,k}}(\underline{f},\ \Delta\hat{\underline{x}}_{k,k}\Delta\underline{x}^T_{k,k})^T\}$$
$$- \frac{1}{4}\,\partial^2_{\hat{\underline{x}}_{k,k}}(f,P_{k,k})\partial^2_{\hat{x}_{k,k}}(f,P_{k,k})^T \tag{3.61}$$

By use of the approximation given in equation (3.49), L_k can be reduced to [2]:

$$L_k = \frac{1}{2}\left(\partial^2_{\hat{\underline{x}}_{k,k}} \underline{f}(\hat{\underline{x}}_{k,k}) P^2_{k,k}\, \partial^2_{\hat{\underline{x}}_{k,k}} f(\hat{\underline{x}}_{k,k})\right) \tag{3.62}$$

Equations (3.60) and (3.62) are the sought recursion for P_k between observations.

The equations for the modified Gaussian second order filter at an observation is given in [2]:

$$\hat{\underline{x}}_{k,k} = \hat{\underline{x}}_{k,k-1} + K_k(z_k - \hat{z}_{k,k-1}) \tag{3.63}$$

$$P_{k,k} = (I-K_kH_k)P_{k,k-1}(I-K_kH_k)^T + K_kW_kK_k^T \tag{3.64}$$

$$K_k = P_{k,k-1}H_k^T[H_kP_{k,k-1}H_k^T + W_k + A_k]^{-1} \tag{3.65}$$

$$A_k = \frac{1}{2}\left(\partial_{\hat{\underline{x}}_{k,k-1}} \underline{g}(\hat{\underline{x}}_{k,k-1}) P^2_{k,k-1}\, \partial^2_{\hat{\underline{x}}_{k,k-1}} g(\hat{\underline{x}}_{k,k-1})\right) \tag{3.66}$$

The necessary equations for the modified Gaussian second order filter are now established, and constitutes of equations (3.58), (3.60) and (3.62) for time updating, and (3.63), (3.59), (3.64) - (3.66) for observation integration.

The equations will now be specialized to the two different representations of the target tracking problem.

3.3.1 Cartesian Coordinate System Model

The Cartesian system model representation of the target motion with bearing only measurements is given by equations (3.3) to (3.7). Since the system dynamics are linear, the second order filter equations for time updating, equations (3.58), (3.60) and (3.62) reduces to the normal Kalman filter equations, given by equations (3.17) and (3.18). The observation equation (3.4), however, is nonlinear, and equations (3.59) and (3.66) have to be evaluated for the nonlinear function $g(x_k)$ given by equation (3.7).

From equations (3.59) and (3.7) we get:

$$\hat{z}_{k,k-1} = \tan^{-1}\left(\frac{\hat{x}_{k,k-1}-x_{sk}}{\hat{y}_{k,k-1}-y_{sk}}\right) + b_k \tag{3.67}$$

where b_k is given by (see equations (3.59) and (3.50)):

$$b_k = \frac{1}{2}\,\partial^2_{\underline{\hat{x}}_{k,k-1}}(\tan^{-1}(\cdot),\ P_{k,k-1}) = \frac{1}{2}\,\mathrm{trace}\left\{\frac{\partial}{\partial \underline{x}}\left[\frac{\partial \tan^{-1}(\cdot)}{\partial \underline{x}}\right]^T\Bigg|_{\underline{x}=\underline{\hat{x}}_{k,k-1}} \cdot P_{k,k-1}\right\} \tag{3.68}$$

Defining (see Fig. 3.1):

$$\left.\begin{aligned} \Delta x &= \hat{x}_{k,k-1} - x_{sk} \\ \Delta y &= y_{k,k-1} - y_{sk} \\ R &= \sqrt{\Delta x^2 + \Delta y^2} \end{aligned}\right\} \tag{3.69}$$

we can calculate $\frac{\partial}{\partial \underline{x}}\left[\frac{\partial \tan^{-1}(\frac{\Delta x}{\Delta y})}{\partial \underline{x}}\right]^T$.

The result:

$$\frac{\partial}{\partial \underline{x}}\left[\frac{\partial \tan^{-1}\left(\frac{\Delta x}{\Delta y}\right)}{\partial \underline{x}}\right]^T = \begin{bmatrix} \frac{2\Delta x\cdot\Delta y}{R^4} & \frac{\Delta x^2-\Delta y^2}{R^4} & 0 & 0 \\ \frac{\Delta x^2-\Delta y^2}{R^4} & \frac{2\Delta x\cdot\Delta y}{R^4} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (3.70)$$

Based on equation (3.70), equation (3.68) becomes

$$b_k = \frac{1}{2\,R^4}\ \text{trace}\begin{bmatrix} -2\Delta x\Delta y\cdot P_{11} + (\Delta x^2-\Delta y^2)P_{21}, & -2\Delta x\Delta y P_{12} + (\Delta x^2-\Delta y^2)P_{22} & 0 & 0 \\ (\Delta x^2-\Delta y^2)P_{11} + 2\Delta x\Delta y\cdot P_{21}, & (\Delta x^2-\Delta y^2)P_{12} + 2\Delta x\Delta y P_{22} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (3.71)$$

Finally:

$$b_k = \frac{1}{2\,R^4}\left[2\Delta x\Delta y(P_{22}-P_{11}) + (\Delta x^2-\Delta y^2)(P_{21} + P_{12})\right] \quad (3.72)$$

Next we have to evaluate equation (3.66) based on equations (3.52) and (3.7). The result is

$$A_k = \frac{1}{R^4}\,[2\Delta x^2\Delta y^2(P_{11}^2+P_{22}^2-P_{12}^2-P_{21}^2) - 2\Delta x\Delta y(\Delta x^2-\Delta y^2)(P_{11}-P_{22})(P_{12}+P_{21})$$
$$+ (\Delta x^2-\Delta y^2)^2(P_{11}P_{22} + P_{12}\cdot P_{21})] \quad (3.73)$$

In equation (3.71) - (3.73) the variables P_{11}, P_{12}, P_{21}, and P_{22} are the position elements of the covariance matrix $P_{k,k-1}$, i.e.,

$$P_{k,k-1} = \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ P_{31} & P_{32} & P_{33} & P_{34} \\ P_{41} & P_{42} & P_{43} & P_{44} \end{bmatrix} \qquad (3.74)$$

The modified Gaussian second order filter for the Cartesian system model representation of the target motion can then be summarized as follows:

Initialization: Equations (3.11) and (3.12).

Observation Integration: Equations (3.13), (3.67), (3.72), (3.65), (3.73) and (3.16).

Time updating: Equations (3.17) and (3.18).

3.3.2 Polar Coordinate System Model

The polar coordinate system representation of our tracking problem is given by equations (3.28) and (3.30). In this case the system dynamics are nonlinear, while the observation equation is linear.

The second order filter equations for time updating, equations (3.58), (3.60) and (3.62) have then to be specialized to our polar coordinate system model, while the observation equations, equations (3.63)-(3.66), (3.59), reduces to the normal Extended Kalman-filter equations, given by equations (3.39)-(3.42).

From equation (3.58) we get:

$$\hat{x}^{P}_{k+1,k} = \underline{f}(\hat{\underline{x}}^{P}_{k,k}) + \frac{1}{2}\underline{c}_k \qquad (3.75)$$

where the vector $\underline{C}_k$ has elements i given by:

$$C_{ki} = \text{trace}\left\{ \frac{\partial}{\partial \underline{x}} \left[\frac{\partial \underline{f}_i}{\partial \underline{x}} \right]^T \Bigg|_{\underline{x}=\hat{\underline{x}}^P_{k,k}} \cdot P_{k,k} \right\} \tag{3.76}$$

Further, we have to calculate the matrix L_k given by equation (3.62).

The calculation of $\underline{C}_k$ and L_k are performed in APPENDIX B. The results are:

$$\begin{bmatrix} C_{1k} \\ C_{2k} \\ C_{3k} \\ C_{4k} \end{bmatrix} = \begin{bmatrix} \sum_{j,i=1}^{4} f1_{ji} P_{ij} \\ \sum_{j,i=1}^{4} f2_{ji} P_{ij} \\ 0 \\ 0 \end{bmatrix} \tag{3.77}$$

$$L_k = \frac{1}{2} \begin{bmatrix} \sum_{j,i=1}^{4} s_{ji} s_{ij} & \sum_{j,i=1}^{4} s_{ji} t_{ij} & 0 & 0 \\ \sum_{j,i=1}^{4} t_{ji} s_{ij} & \sum_{j,i=1}^{4} t_{ji} t_{ij} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.78}$$

The elements of the matrices F1, F2, S and T are given in APPENDIX B.

The modified Gaussian second order filter for the Polar Coordinate system model of the tracking problem can then be summarized as follows:

Initialization: Equations (3.37) and (3.38)

Observation integration: Equations (3.39)-(3.42).

Time updating: Equations (3.75), (3.77), (3.60) and (3.78).

3.4 Iterated Extended Kalman Filter [2]

The following 5 approaches to solve the target tracking problem are all approaches involving some sort of iteration scheme. The first approach, called the iterated extended Kalman-filter, is a local iteration scheme, and is designed in order to reduce the effect of measurement function nonlinearity. This approach can therefore be tried on the Cartesian coordinate system representation, which have linear system dynamics and nonlinear measurement model.

Since this approach is very well documented in Jazwinski [2], pp. 278-279, it should be unnecessary to repeat the equations in this text. For completeness, however, the approach is included in APPENDIX C.

3.5 Iterated Linear Filter-Smoother [2]

If we have system dynamics nonlinearities, the preceding iterator will not improve the estimate $\hat{\underline{x}}_{k+1,k}$ due to system nonlinearities acting on the interval [kT, (k+1)T].

The preceding iterator can therefore not be used in connection with the polar coordinate system representation of the target motion.

The "Iterated Linear Filter Smoother", proposed by Jazwinski [2], pp. 279-281, include the time updating process in the iteration loop. This approach can therefore be tried on the polar coordinate system representation of our tracking problem.

Again, since the approach is well documented in [2], the iterator equations are exiled to APPENDIX D.

3.6 Global Iterated Filters

As we mentioned in Section 2, one of the main problems with bearing only tracking is the poor degree of observability. The observability of range is entirely dependent on the maneuvering scheme of the observer.

In order to improve the observability, and at the same time improve the reference trajectory and thereby get a more optimal utilization of each observation, a number of global iteration schemes are proposed. The iterations will here not be performed at the time kT (as in section 3.4) or over the time interval [kT, (k+1)T] (as in section 3.5), but over the greater time interval [(k-N)T, kT], where N is some integer constant, may be variable.

This iterator requires, however, a lot of memory capacity for storing of variables. The following sequences has to be stored (M is an integer constant $\geq$ N):

Observation Sequence:

$$Z_k = \{z_{k-M}, z_{k-M+1}, z_{k-M+2}, \ldots, z_k\} \qquad (3.79)$$

Observer position, velocity and acceleration:

$$X_{sk} = \{\underline{x}_{s_{k-M}}, \underline{x}_{s_{k-M+1}}, \ldots, \underline{x}_{sk}\} \tag{3.80}$$

$$V_{sk} = \{\underline{v}_{s_{k-M}}, \underline{v}_{s_{k-M+1}}, \ldots, \underline{v}_{sk}\}$$

$$A_{sk} = \{\underline{a}_{s_{k-M}}, \underline{a}_{s_{k-M+1}}, \ldots, \underline{a}_{sk}\} \tag{3.82}$$

covariance matrix:

$$P_k = \{P_{k-M,k-M}, \ldots, P_{k,k}\} \tag{3.83}$$

As we can see, the memory capacity requirements will depend on the size of the integer M. Also, if the iteration time interval is fixed (N = constant) or the variation space for N is fixed ($N_{low} \leq N \leq N_{HIGH}$, where N_{LOW} and N_{HIGH} are constants), the requirements for storing of the covariance matrix can be restricted to

$$P_k = \{P_{k-N_{HIGH}}, P_{k-N_{HIGH}+1}, \ldots, P_{k-N_{LOW}}\} \tag{3.84}$$

The requirements concerning storing of the observer's velocity and acceleration will also depend on the representation of the tracking problem. Equations (3.79)-(3.83) therefore represent an upper bound on the memory capacity requirements.

Intuitively, if the observer has performed a maneuvre during the time interval [(k-N)T, kT], a global iteration scheme will improve the observability. The idea is that we then get crossbearings over this time interval.

Three different approaches to the Global Iterated Filter will be outlined in the following.

3.6.1 Cartesian Coordinate System Representation

The first approach to be Global Iterated Filter utilize the Cartesian Coordinate system representation described in Section 3.1.

The number of iterations, i, on each set of data, has to be limited. The following test is proposed in order to stop the iteration sequence at a time instant kT:

$$|\underline{\hat{x}}^{i+1}_{k,k} - \underline{\hat{x}}^{i}_{k,k}| \leq \underline{\varepsilon} \tag{3.85}$$

where the $\underline{\varepsilon}$ vector has to be selected through simulations as a compromise between accuracy and computertime (number of iterations).

If a target maneuvre is detected, the size of the iteration interval should be decreased in order to not perform iterations on premaneuvre target data. The iterator should therefore have an adaptive calculation sequence for N, in connection with the maneuvre detection system. We will return to this point when dealing with the maneuvre detection system in Chapter 5.

The iterator will work in the following way:

Having performed iterations on the observation sequence $\{z_{k-2N}, z_{k-2N+1},\ldots,z_{k-N}\}$, until the criterium $|\underline{\hat{x}}^{i}_{k-N,k-N}-x^{i-1}_{k-N,k-N}| \leq \underline{\varepsilon}$ is satisfied, we have the state vector $\underline{\hat{x}}_{k-N,k-N}$ and the covariance matrix $P_{k-N,k-N}$. These are stored in the computer memory. Perform the global iteration algorithm subsequently on the observation sequences $\{z_{k-2N+1},\ldots, z_{k-N+1}\}$, $\{z_{k-2N+2},\ldots, z_{k-N+2}\}$ etc. At each fulfilled iteration, say at samplepoint jT, the covariance matrix $P_{j,j}$ and the state vector $\underline{\hat{x}}_{j,j}$ are stored. Like in the conventional Kalman-filter case, however, only the value of the state vector at the last samplepoint is needed, so the next state vector $\hat{\underline{x}}_{j,j}$ can use the same storage are as $\hat{\underline{x}}_{j-1,j-1}$ (and thus destroy $\hat{\underline{x}}_{j-1,j-1}$).

Eventually, when we arrive at sample kT, we have the following data base picture (older data have been discarded):

Observation sequence: $\{z_{k-N},\ldots,z_k\}$.

Covariance sequence: $\{P_{k-N,k-N},\ldots, P_{k-1,k-1}\}$

Observer position sequence: $\{\underline{x}_{s_{k-N}},\ldots,\underline{x}_{sk}\}$

State vector: $\hat{\underline{x}}_{k-1,k-1}$

From this startpoint the following calculations are performed:

1. Time updating. Calculate $\hat{\underline{x}}_{k,k-1}$ and $P_{k,k-1}$.

2. Observation integration. Process z_k, resulting in $\hat{\underline{x}}^i_{k,k}$ and $P^i_{k,k}$, where i=1.

3. Timebackdating:

$$\underline{x}^i_{k-N,k-N} = \phi(-NT)\, \hat{\underline{x}}^i_{k,k} \tag{3.86}$$

$P_{k-N,k-N}$ = fetched from the data base

4. Reprocess the observation sequence, resulting in $\hat{\underline{x}}^{i+1}_{k,k}$, $P^{i+1}_{k,k}$.

5. Perform the test described by equation (3.85) to decide whether to continue the iteration loop, or to stop the iterations on this observation sequence.

a. If decision is continue the iterations, set i=i+1, and start from step 3 again.

b. If decision is to stop the iterations, store $\hat{\underline{x}}^{i+1}_{k,k}$ and $P^{i+1}_{k,k}$ in the computer memory, set k = k+1, and start from step 1 again.

This iterator will successively improve the linearization trajectory for H_k (given by equation (3.10)). The utilization of the data sequence therefore will be increasingly more optimal.

3.6.2 Polar Coordinate System Representation

The second approach to the Global Iterated Filter utilize the Polar Coordinate system representation described in section 3.2.

The mechanization of the iterator is identical to the iterator in sequence 3.6.1, except for the time-back dating step.

Due to the nonlinearity of the system dynamics, where the target position information is given in relative coordinates (relative to the observer), timeback dating by use of the inverse form of equation (3.25) (with the process noise vector $\underline{v}_k = 0$), would be unnecessary time-consuming. $\underline{x}^{P}_{i,i}$ would then have to be calculated for decreasing i, i=k, k-1,...,k-N, not in one big backdating step as we could do for the Cartesian Coordinate system model (see equation (3.86)).

In order to save computertime, the following approach for time-back-dating is proposed:

1. The state vector's position information is transformed to Cartesian representation through:

$$\hat{x}_{k,k} = R_{k,k} \cdot \sin \phi_{k,k} + x_{sk} \tag{3.87}$$

$$\hat{y}_{k,k} = R_{k,k} \cos \phi_{k,k} + y_{sk} \tag{3.88}$$

where the different variables are defined (without time subscript k, though) on Fig. 3.1. Since the velocity representation is equal in the two representations, we now have a complete, Cartesian state vector $\hat{\underline{x}}^{c}_{k,k}$.

2. Perform time backdating through equation (3.86), resulting in $\hat{\underline{x}}^{c}_{k-N,k-N}$.

3. Perform the transformation from Cartesian to Polar representation through the equations

$$\hat{\phi}_{k-N,k-N} = \tan^{-1}\left(\frac{\hat{x}_{k-N,k-N} - x_{s_{k-N}}}{\hat{y}_{k-N,k-N} - y_{s_{k-N}}}\right) \tag{3.89}$$

$$\hat{R}_{k-N,k-N} = \sqrt{(\hat{x}_{k-N,k-N} - x_{s_{k-N}})^2 + (\hat{y}_{k-N,k-N} - y_{s_{k-N}})^2} \tag{3.90}$$

Now, if the time back dating step of the iterator described in section 3.6.1 (step no. 3) were changed to include processing of equations (3.87) and (3.88) before processing of equations (3.86), and to include processing of equations (3.89) and (3.90) subsequent to equation (3.86), the calculation steps 1-5 given in section 3.6.1 will be valid also for the Polar coordinate system case.

One additional preferable requirement exists, however: The database should also contain the observer's incremental position change from sample to sample, given by the left hand side of equation (3.24), over the time interval [(k-N)T, kT].

Alternatively, these quantities can be calculated from the sequences given in equations (3.81) and (3.82), if these are in the database.

This iterator will successively improve the linearization trajectory for F_k (given by equations (3.31)-(3.32)). The computational requirements will, however, exceed the iterator in Section 3.6.1.

3.6.3 "Serial" filters

If the two previous described global iterators process each observation sequence $\{z_{k-N}, z_{k-N+1}, \ldots, z_k\}$ twice, which is the minimum number of iterations, each individual observation z_k will be processed 2N times. It is thus obvious that these iterators will increase the computational burden very heavily, as compared to the Extended Kalman-filter case.

In an attempt to reduce the calculation load, and still bring along the advantages of the global iteration approach, a different iterator ...oposed for the Cartesian system representation. The effect of this ... in fact two coupled filters, running along the target ...ory with a certain time delay, NT, between the filters. This is the reason for the name serial in the heading of this section.

The two serial filters are added together in one system description, in order to get a compact form. We have:

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ v_{x\,k+1} \\ v_{y\,k+1} \\ x_{k-N+1} \\ y_{k-N+1} \\ v_{x\,k-N+1} \\ v \end{bmatrix} = \left[\begin{array}{cccc|cccc} 0 & 0 & 0 & 0 & 1 & 0 & T(N+1) & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & T(N+1) \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \hline 1 & 0 & -(N-1)T & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -(N-1)T & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ & & & & & & & \end{array}\right] \cdot \begin{bmatrix} x_k \\ y_k \\ v_{xk} \\ v_{yk} \\ x_{k-N} \\ y_{k-N} \\ v_{xk-N} \\ \end{bmatrix} +$$

$$\left[\begin{array}{cc|cc} T^2/2 & 0 & 0 & 0 \\ 0 & T^2/2 & 0 & 0 \\ T & 0 & 0 & 0 \\ 0 & T & 0 & 0 \\ \hline 0 & 0 & T^2/2 & 0 \\ 0 & 0 & 0 & T^2/2 \\ 0 & 0 & T & 0 \\ 0 & 0 & 0 & T \end{array}\right] \cdot \begin{bmatrix} v_{1k} \\ v_{2k} \\ v_{1k-N} \\ v_{2k-N} \end{bmatrix} \tag{3.91}$$

$$\begin{bmatrix} z_k \\ z_{k-N} \end{bmatrix} = \begin{bmatrix} \tan^{-1}\left(\dfrac{x_k - x_{sk}}{y_k - y_{sk}}\right) \\ \tan^{-1}\left(\dfrac{x_{k-N} - x_{s_{k-N}}}{y_{k-N} - y_{s_{k-N}}}\right) \end{bmatrix} + \begin{bmatrix} w_k \\ w_{k-N} \end{bmatrix} \tag{3.92}$$

In equations (3.91) and (3.92), N is the number of samples between the two parts of the state vector.

We now define:

$$\underset{\sim}{\tilde{x}}_k = \begin{bmatrix} \underline{x}_{1k} \\ \underline{x}_{2k} \end{bmatrix} = \begin{bmatrix} \underline{x}_k \\ \underline{x}_{k-N} \end{bmatrix} \tag{3.93}$$

and

$$\tilde{v}_k = \begin{bmatrix} \underline{v}_{1k} \\ \underline{v}_{2k} \end{bmatrix} = \begin{bmatrix} \underline{v}_k \\ \underline{v}_{k-N} \end{bmatrix} \tag{3.94}$$

Then, equation (3.91) can be written:

$$\underset{\sim}{\tilde{x}}_{k+1} = \left[\begin{array}{c|c} 0 & \phi((N+1)T) \\ \hline \phi(-(N-1)T) & 0 \end{array}\right] \underset{\sim}{\tilde{x}}_k + \left[\begin{array}{c|c} \theta(T) & 0 \\ \hline 0 & \theta(T) \end{array}\right] \underset{\sim}{\tilde{v}}_k \tag{3.95}$$

Further defining:

$$\tilde{\underline{z}}_k = \begin{bmatrix} z_{1k} \\ \\ z_{2k} \end{bmatrix} = \begin{bmatrix} z_k \\ \\ z_{k-N} \end{bmatrix} \tag{3.96}$$

$$\tilde{\underline{g}}(\tilde{\underline{x}}_k) = \begin{bmatrix} \tan^{-1}\left(\dfrac{\Delta x_k}{\Delta y_k}\right) \\ \tan^{-1}\left(\dfrac{\Delta x_{k-N}}{\Delta y_{k-N}}\right) \end{bmatrix} \tag{3.97}$$

and

$$\tilde{\underline{w}}_k = \begin{bmatrix} w_k \\ w_{k-N} \end{bmatrix} \tag{3.98}$$

the observation equation (3.92) can be written:

$$\tilde{\underline{z}}_k = \tilde{\underline{g}}(\tilde{\underline{x}}_k) + \tilde{\underline{w}}_k \tag{3.99}$$

Lastly, we define

$$\tilde{H}_k = \left.\frac{\partial \tilde{\underline{g}}}{\partial \tilde{\underline{x}}_k}\right|_{\tilde{\underline{x}}_k = \hat{\tilde{\underline{x}}}_{k,k-1}} = \left[\begin{array}{cccc|cccc} \frac{\Delta y}{R^2}(k) & -\frac{\Delta x}{R^2}(k) & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & \frac{\Delta y}{R^2}(k-N) & \frac{-\Delta x}{R^2}(k-N) & 0 & 0 \end{array}\right] \tag{3.100}$$

$$\tilde{V}_k = \begin{bmatrix} V_k & 0 \\ \\ 0 & V_{k-N} \end{bmatrix} \tag{3.101}$$

$$\tilde{W}_k = \begin{bmatrix} W_k & 0 \\ \\ 0 & W_{k-N} \end{bmatrix} \tag{3.102}$$

$$\tilde{\phi}(T) = \left[\begin{array}{c|c} 0 & \phi((N+1)T) \\ \hline \phi(-(N-1)T) & 0 \end{array}\right] \tag{3.103}$$

and

$$\tilde{\theta}(T) = \left[\begin{array}{c|c} \theta(T) & 0 \\ \hline 0 & \theta(T) \end{array}\right] \tag{3.104}$$

where V_k and W_k are given by equations (3.8) and (3.9), $\theta(T)$ by equation (3.6) and $\phi(T)$ by equation (3.5). $\tilde{H}_k$ can also be written:

$$\tilde{H}_k = \left[\begin{array}{c|c} H_k & 0 \\ \hline 0 & H_{k-N} \end{array}\right] \tag{3.105}$$

where H_k is given by equation (3.10).

The Extended Kalman filter equations for this augmented $(\tilde{\cdot})$ system, are given by equations (3.11)-(3.18), if replacement with the augmented variables are performed in the equations.

In order to start up the augmented system correctly, the first N observations are processed with the Kalman-filter given in section 3.1. Then the initial values for the augmented system will be:

$$\tilde{\underline{x}}_{0,-1} = \begin{bmatrix} \hat{\underline{x}}_{k,k} \\ \underline{x}_0 \end{bmatrix} \tag{3.106}$$

$$\tilde{P}_{0,-1} = \begin{bmatrix} P_{k,k} & 0 \\ 0 & P_0 \end{bmatrix} \tag{3.107}$$

At this point it is appropriate with a few comments on this approach to global iteration. As we can see from equation (3.95), the first part of the state vector, $\underline{x}_{1k}$, is updated on the basis of the second part of the state vector, $\underline{x}_{2k}$, and vice versa. From equation (3.92) we can further deduce that each observation will be integrated twice.

Looking at the iteration scheme given in section 3.6.1, we can see that our "serial" filter is equivalent with the steps 1-4, with step 4 reduced to integration of z_{k-N} followed by a time updating step up to time (k+1)T. Step 5 does not exist, each observation is only reprocessed once.

The global iteration effect of this approach should then have been demonstrated.

It should be pointed out that this scheme can be augmented to incorporate a selected number of filters, say M, following each other with a time difference NT. The resulting model would then look like:

$$\begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \\ \vdots \\ \underline{x}_M \end{bmatrix}(k+1) = \begin{bmatrix} 0 & \phi((N+1)T) & & 0 \\ 0 & \ddots & \ddots & \\ \vdots & & \ddots & \phi((N+1)T) \\ \phi(-(MN-1)T) & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} \underline{x}_1 \\ \vdots \\ \underline{x}_M \end{bmatrix}(k) \cdot \begin{bmatrix} \theta(T) & & 0 \\ & \ddots & \\ 0 & & \theta(T) \end{bmatrix} \begin{bmatrix} \underline{v}_1 \\ \vdots \\ \underline{v}_M \end{bmatrix}(k) \tag{3.108}$$

with an observation equation:

$$\begin{bmatrix} z(k) \\ \vdots \\ z(k-MN) \end{bmatrix} = \begin{bmatrix} \tan^{-1}\left(\frac{\Delta x}{\Delta y}(k)\right) \\ \vdots \\ \tan^{-1}\left(\frac{\Delta x}{\Delta y}(k-MN)\right) \end{bmatrix} + \begin{bmatrix} w(k) \\ \vdots \\ w(k-MN) \end{bmatrix} \tag{3.109}$$

In this case, each observation will be processed M times.

3.7 Parallel Filter Approach

The last approach that will be explored is the parallel filter approach. In this case, where we have M parallel filters with different linearization trajectories, only the simplest Cartesian model for each filter will be considered, in order to reduce the computer time. Each filter will, therefore, be represented by the mathematical model given by equations (3.3)-(3.7).

The philosophy behind this approach is the following: Each filter will be initialized with different range, R_{oi}, i=1,2,...,M. The initial values for the velocity components for each filter will depend on R_0, if the initialization routine proposed in Chapter 4 is used.

In the time interval from initialization until the first observer maneuvre, target range is unobservable. The utilization of the bearing observations in this time interval will be nonoptimal for all the filters except the one(s) with approximately correct range. When the observer performs a maneuvre, target range becomes observable. It is then possible to identify which of the M filters that has the most correct range.

The resulting state vector for this filtering scheme can be given by [14]:

$$\hat{\underline{x}}_k = \sum_{i=1}^{M} P(R_{ki} = R_k/z_k) \cdot \hat{\underline{x}}_{ki} \qquad (3.110)$$

where the probabilities $P(R_{ki} = R_k/z_k)$, i=1,...,M, have to be calculated. The question is how.

Now, the most likely quantity to contain information about these probabilities, are the innovation sequences for each filter. These are given by:

$$\varepsilon_{ki} = z_k - \hat{z}_{k,k-1_i}, \; i=1,2,\ldots,M \tag{3.111}$$

In our special case, ε_{ki} is a scalar variable. Under a number of conditions, which include the requirement of equality between the physical system and the mathematical model contained in the Kalman-filter, the innovation sequence has been shown to be a white Gaussian sequence with statestics $\sim N(0, \sigma_{ki}^2)$, where σ_{ki}^2 is given by:

$$\sigma_{ki}^2 = H_{k,k-1}^i P_{k,k-1}^i H_{k,k-1}^{iT} + W_k \tag{3.112}$$

σ_{ki}^2, i=1,2,...,M, are already calculated by the Kalman filter algorithm. $m_{ki} = 0$, and σ_{ki}^2, i=1,2,...,M, represents the expected mean and variance for the innovation sequence.

The expected variance of the innovation sequence given by equation (3.112) will be different for each filter, i.e., the variance will depend on R_i. Equation (3.112) can be written:

$$\sigma_{ki}^2 = \frac{1}{R_i^2} [P_{11_{k,k-i}}^i \cdot \cos^2\phi_{k,k-1}^i + P_{22_{k,k-1}}^i \cdot \sin^2\phi_{k,k-1}^i - P_{12_{k,k-1}}^i \cdot \sin 2\phi_{k,k-1}^i] + w_k$$

or equivalently (3.113)

$$\sigma_{ki}^2 = \frac{1}{R_i^2} [P_{11_{k,k-1}}^i + P_{22_{k,k-1}}^i - \sigma_{R_i}^2] + w_k \tag{3.114}$$

where

$$\sigma^2_{R_i} = P^i_{11_{k,k-1}} \sin^2\phi^i_{k,k-1} + P^i_{22_{k,k-1}} \cdot \cos^2\phi^i_{k,k-1} + P^i_{12_{k,k-1}} \cdot \sin 2\phi^i_{k,k-1} \tag{3.115}$$

The difference between the M different ϕ_i's will not be significant in this case, since the observability of the bearing, ϕ, is very high.

The values of the covariance matrix elements, P_{11}, P_{22} and P_{12}, however, will depend on R_i. In the equations for calculation of $P_{k,k}$, the observation noise will be weighted by R_i^2. See Appendix E for detailed derivation of the covariance equations.

As can be seen from Appendix E, each of the elements of the covariance matrix $P_{k,k}$ will increase for increasing R_i. The value of the expected innovation variance given by equation (3.113) will, therefore, not decrease as a function of R_i^{-2}, however, σ^2_{ki} will decrease as some function of R_i, since the elements of $P_{k,k}$ are limited in their growth, and does not increase with the power of R_i^2 anyway. A detailed simulation analysis of the equations given in Appendix E has to be carried out, in order to reveal the exact behaviour of $P_{k,k}$ as a function of R_i.

The actual statistics for each filter i, i=1,2,...,M, can be approximated by:

$$m_{\varepsilon_{ki}} = \frac{1}{N+1} \sum_{j=k-N}^{k} \varepsilon_{ji} \tag{3.116}$$

$$\sigma^2_{\varepsilon_{ki}} = \frac{1}{N+1} \sum_{j=k-N}^{k} (\varepsilon_{ji} - m_{\varepsilon_{ji}})^2 \tag{3.117}$$

The idea is now to compare the innovation sequence's actual statistics with its expected statistics, and thereby get an expression for the

Probabilities $p(R_{ki} = R_k/z_k)$.

In doing this, we have to consider the following two cases:

1. System not observable in the time interval $[(k-N)T, kT]$.
2. System is observation in the time interval $[(k-N)T, kT]$.

By observable or not we mean whether or not the range/velocity ambiguity can be resolved through the bearing observation over the time interval $[(k-N)T, kT]$. In any case, the bearing to target will be observable, and so will the ratio $\Delta v/R$, where Δv is the relative velocity between target and observer.

The information as to whether the system is observable or not, and the degree of the observability, is known to the tracking routine, since the observer's maneuvering history in the time interval $[(k-N)T, kT]$ is known.

3.7.1 <u>System not observable.</u>

If the observer's velocity- and/or course-changes are below defined thresholds over the time interval in question, we know that the range/velocity ambiguity can not be resolved from bearing observations only.

This case can further be devided into the following two sub-cases:

1. The target is not maneuvering during the time interval.
2. The target is maneuvering during the time interval.

A maneuver is defined as a definite course and/or velocity change, not the natural small fluctuations in velocity and course due to waves, wind, etc.

3.7.1.1 Target not Maneuvering

If the target is moving along a straight course with constant speed, the actual mean value of the innovation signal given by equation (3.116) will be close to zero (below a defined threshold, say m_{Lt}).

Since we know that the target's natural fluctuations about some mean course and speed doesn't depend on the distance from where it is observed, it is likely that the bearing statistics should contain some range information.

In fact, this case looks like the stationary process case where adaptive noise estimation is possible. See Mehra [15], [16], and Chin [17], [18].

In this case we propose to use the covariance matching method, however, not to do any adaption (even if that should also be possible), but to arrive at an expression for the probability function $p(\hat{R}_i = R/z_k)$.

If we set the equations (3.114) and (3.117) equal (expected variance = actual variance), we get:

$$\frac{1}{R_i^2}\left(P^i_{11_{k,k-1}}(R_i) + P^i_{22_{k,k-1}}(R_i) - \sigma^2_{R_i}\right) + W_k = \sigma^2_{\varepsilon_{k_i}} \tag{3.118}$$

Solving this equation for the only explictly occurring R_i in the equation gives us:

$$R_i^* = \sqrt{\frac{P^i_{11_{k,k-1}}(R_i) + P^i_{22_{k,k-1}}(R_i) - \sigma^2_{R_i}}{\sigma^2_{\varepsilon_{k_i}} - W_k}} \tag{3.119}$$

where R_i^* may be different from R_i. The difference:

$$\Delta R_i = R_i^* - R_i, \; i=1,2,\ldots,M \tag{3.120}$$

will be an expression for the range error for each filter.

We propose the following form for the probability function $p(\hat{R}_i = R/z_k)$:

$$p(R_i = R/z_k) = K \cdot e^{-\alpha \cdot \Delta R_i^2} \tag{3.121}$$

where the variables K and α will depend on the process noise covariance matrix V_k, and have to be decided upon through simulations.

3.7.1.2 Target Maneuvering

During a target maneuvre, the method described in section 3.7.1.1 will not work, since we in this case gets temporary changes of unknown size and duration in the process noise covariance matrix V_k, that are not taken into account in the calculation of the expected variance of the innovation signal.

Therefore, when a target maneuvre is detected, for example by an abrupt change in both the actual mean and variance of the innovation signal, the last calculated probabilities $p(R_i = R/z_k)$ should be used, until the filters have adapted to the new course and/or speed, and $m_{\varepsilon_{ki}}$ again drops below the threshold m_{Lt}.

Alternatively, if it is possible to decide upon the time delay between the instant of the target maneuvre and its detection, the probabilities $p(R_i = R/z_{kT-\Delta t})$ should be used, where Δt is the described time delay.

Target maneuvre detection will be treated in Chapter 5.

3.7.2 System Observable

If the observer is maneuvering during the time interval $[(k-N)T, kT]$, the range/velocity ambiguity can be resolved based on the bearing observations only. In this case the filter with the most correct range can be identified, if the target is not maneuvering during the time interval in question.

We therefore have to consider the same two sub-cases as in section 3.7.1:

1. The target is not maneuvering during the time interval.
2. The target is maneuvering during the time interval.

3.7.2.1 Target not maneuvering

The filter with correct range will not change its actual statistics when the observer performs a maneuvre. This fact can be utilized to prune off the filters with wrong range.

By monitoring m_{ε_k} and $\sigma^2_{\varepsilon_{ki}}$ given by equations (3.116) and (3.117) through an observer maneuvre, it is thus possible to identify the filter with the best range as the filter with the least change in actual statistics.

In this case, we continue to calculate the probability functions given by equation (3.121) for a gradually decreasing number of parallel filters, as the filters with the wrongest ranges successively will be removed from being in an active state to a so-called "dormant" state, where they no longer are updated.

The selection of the filters with bad ranges can be made from monitoring $m_{\varepsilon_{ki}}$. When $m_{\varepsilon_{ki}}$ exceeds a given threshold m_{Ht}, the filter no. i is temporarily pruned off the ensemble.

In this way the number of necessary parallel filters can be reduced, and if the observer maneuvre extends over a time interval long enough, or if the observer performs successive maneuves while the target remains on straight course with constant speed, we will eventually be left with one filter, with correct range.

This situation will prevail only until a target maneuvre is detected, when all the M parallel filters should be initialized again.

3.7.2.2 Target maneuvering

When a target maneuvre is detected, all the M filters should be re-initialized, independent of the number of active filters at the time of maneuvre detection.

The M filters should be reinitialized with different ranges R_{oi} centered about the estimated range $\hat{R}_{k,k}$, given by the state vector $\hat{\underline{x}}_{k,k}$ at the time of maneuvre detection. The difference between the individual R_{0i}'s should be much less than during the first initialization, since range is not at all so uncertain this time.

Simultaneously, the velocity elements of the covariance matrix should be increased, in order to allow the filter to adapt to the targets new course and speed.

After reinitialization, the calculation of the probability function given by equation (3.121) is resumed.

Target maneuvre detection and handling in connection with parallel filters are treated in section 5.3.

3.7.3 Closing remarks, parallel filters.

The parallel filter approach seems very promising, however, the calculation load imposed by this approach will be substantial.

In order to arrive at a practical parallel filter solution, the idea of adaptively reducing the number of parallel filters have to be investigated thoroughly. This can only be done through a simulation study.

Further, this approach claim for a substantial simulation study in order to "tune" in the different thresholds, time intervals and variables defined in the previous sections.

One fundamental difference between the parallel filter approach and the single filter approach, is the range covariance. For the parallel filter approach the range covariance should be kept low, in order to make each filter "stiff" in range. This is not normally the case for the single filter, where range covariance has to be high in order to allow the filter to adopt the correct range during the periods when range is observable.

4. INITIALIZATION ROUTINE

4.1 General

Bearing only tracking from a single, moving observer is, in the initialization phase, very dependent on the selected initial values of range, course and speed of the target.

The initial values of the elements of the covariance matrix for the estimation error have also an important influence on the initial track.

One method commonly used for selecting initial data for bearing only trackers, is to utilize the knowledge of range reach of the bearing sensor. The argument is the following: If a target is detected at some time T_D, that could not be "seen" by the sensor at times $< T_D$, the reason is that the target has just entered the reach area of the sensor, and is opposing the observer. The initial range is therefore given by the sensor specifications, and the initial course is towards the observer (Initial velocity still has to be picked out of the air).

Since the range reach of the sensor usually depends on the environmental conditions, this method frequently will give very bad initial data.

The purpose of this chapter is to derive an alternative method, where initial data can be selected in a more optimal manner, based on calculations performed on a given set of bearing observations from the moving observer, and the position history of the observer.

The following derivation will also give us some valuable insight in the observability problem, and show the conditions under which the range/velocity ambiguity can be resolved based on bearing observations only.

4.2 Geometric Problem Visualization

A typical geometric situation is depicted in Fig. 4.1.

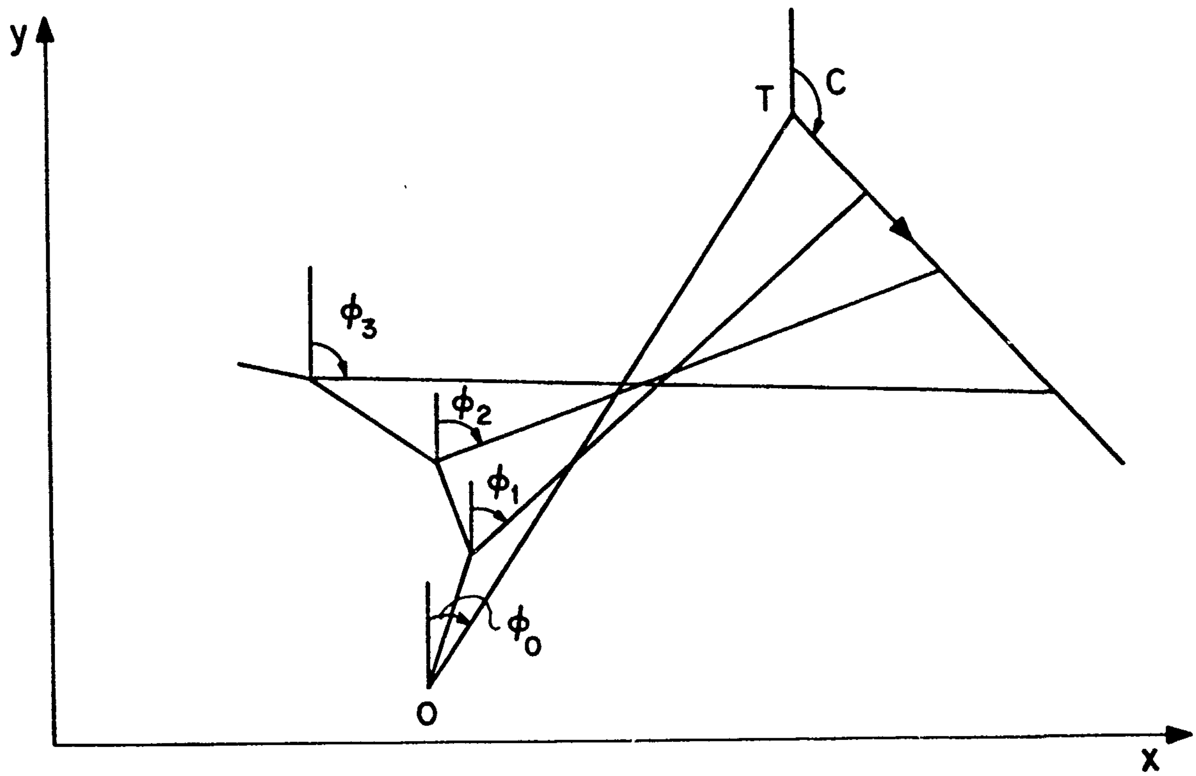


Fig. 4.1 Geometrical Situation

A moving observer, 0, observes a number of bearings to the target, T, at the time points t_0, t_1, t_2,..., where the time difference between any two consecutive bearings may be different.

The following assumptions are made:

1. The target is moving with constant course and speed.
2. The observer's velocity and course are constant between the observations. (This restriction will be removed later).
3. The bearings are noise free.

Assumption number 3 is obviously not true in reality. However, by use of this assumption it is possible to arrive at certain results.

In the discussion of these results later in this chapter, methods for reducing the effect of this assumption being violated will be suggested.

4.3 Calculation of target velocity components.

Based on three consecutive bearings ϕ_0, ϕ_1, ϕ_2, and an assumed start range, R_0, the target velocity components can be calculated. We have the follow. three equations (see also equation (3.25)):

$$\frac{\Delta L_N(\Delta t_1)}{R_0+\Delta L_p(\Delta t_1)} = \tan(\phi_1-\phi_0) \tag{4.1}$$

$$R_1 = \sqrt{(R_0+\Delta L_p(\Delta t_1))^2 + (\Delta L_N(\Delta t_1))^2} \tag{4.2}$$

$$\frac{\Delta L_N(\Delta t_2)}{R_1 + \Delta L_p(\Delta t_2)} = \tan(\phi_2-\phi_1) \tag{4.3}$$

where

$$\Delta t_1 = t_1-t_0 \tag{4.4}$$

$$\Delta t_2 = t_2-t_1 \tag{4.5}$$

or generally

$$\Delta t_i = t_i-t_{i-1}, \; i=1,2 \tag{4.6}$$

and

$$\Delta L_N(\Delta t_i) = \cos \phi_{i-1} [v_x\Delta t_i-\Delta x_{s_{i-1}}] - \sin \phi_{i-1}[v_y\Delta t_i - \Delta y_{s_{i-1}}] \tag{4.7}$$

$$\Delta L_p(\Delta t_i) = \sin \phi_{i-1}[v_x \Delta t_i - \Delta x_{s_{i-1}}] + \cos \phi_{i-1}[v_y \Delta t_i - \Delta y_{s_{i-1}}] \tag{4.8}$$

In equations (4.7) and (4.8), assumptions 1 and 2 from section 4.2 are used. The target movement along x- and y-directions in the time interval Δt_i are given by $v_x \Delta t_i$ and $v_y \cdot \Delta t_i$ respectively (no acceleration term), while the observer movement along the same axis in the time interval are given by:

$$\Delta x_{s_{i-1}} = v_{sx_{i-1}} \cdot \Delta t_i + \frac{1}{2} a_{sx_{i-1}} \Delta t_i^2 \tag{4.9}$$

$$\Delta y_{s_{i-1}} = v_{sy_{i-1}} \Delta t_i + \frac{1}{2} a_{sy_{i-1}} \Delta t_i^2 \tag{4.10}$$

If the observer position increments in the time interval Δt_i is calculated on the basis of equations (4.9) and (4.10), assumption 2 is necessary. However, by deviding the time interval Δt_i into smaller intervals, and calculating $\Delta x_{s_{i-1}}$ and $\Delta y_{s_{i-1}}$ as a summation of position increments over these intervals, provided that velocity and acceleration data for the observer in these intervals exists, the assumption of constant velocity and course can be changed to yield arbitrarily small time increments.

Now, from equation (4.1) we have:

$$\Delta L_N(\Delta t_1) = \tan(\phi_1 - \phi_0)(R_0 + \Delta L_p(\Delta t_1)) \tag{4.11}$$

Inserting from equation (4.11) into equation (4.2) gives:

$$R_1 = (R_0 + \Delta L_p(\Delta t_1)) \cdot \sqrt{1 + \tan^2(\phi_1 - \phi_0)} \tag{4.12}$$

Then:

$$R_1 = (R_0 + \Delta L_p(\Delta t_1))/\cos(\phi_1-\phi_0) \tag{4.13}$$

Further:

$$R_1 = \frac{R_0+\sin\phi_0[v_x\Delta t_1-\Delta x_{s_0}] + \cos\phi_0[v_y\Delta t_1-\Delta y_{s_0}]}{\cos(\phi_1-\phi_0)} \tag{4.14}$$

Inserting for $\Delta L_N(\Delta t_1)$ and $\Delta L_p(\Delta t_1)$ in equation (4.1), gives, after some calculation:

$$v_x = \frac{1}{\Delta t_1}\Delta x_{s_0} + \tan\phi_1(v_y - \frac{1}{\Delta t_1}\Delta y_{s_0}) + \frac{R_0}{\Delta t_1}\frac{\sin(\phi_1-\phi_0)}{\cos\phi_1} \tag{4.15}$$

Now, inserting for v_x from equation (4.15) in equation (4.14) gives:

$$R_1 = \frac{R_0\cos\phi_0 + (v_y\Delta t_1-\Delta y_{s0})}{\cos\phi_1} \tag{4.16}$$

We also insert for v_x from equation (4.15) in the equations for $\Delta L_N(\Delta t_2)$ and $\Delta L_p(\Delta t_2)$ (equations (4.7) and (4.8), with i=2). The results are:

$$\Delta L_N(\Delta t_2) = \frac{\Delta t_2}{\Delta t_1}R_0 \cdot \sin(\phi_1-\phi_0) + \cos\phi_1\left(\frac{\Delta t_2}{\Delta t_1}\Delta x_{s_0}-\Delta x_{s_1}\right.$$
$$\left. - \sin\phi_1\left(\frac{\Delta t_2}{\Delta t_1}\Delta y_{s0} - \Delta y_{s1}\right)\right) \tag{4.17}$$

$$\Delta L_p(\Delta t_2) = \frac{\Delta t_2}{\Delta t_1}R_0\tan\phi_1\cdot\sin(\phi_1-\phi_0) + \sin\phi_1\left(\frac{\Delta t_2}{\Delta t_1}\Delta x_{s0}-\Delta x_{s1}\right)$$
$$+ \frac{\Delta t_2}{\cos\phi_1}v_y - \frac{1}{\cos\phi_1}\left(\frac{\Delta t_2}{\Delta t_1}\Delta y_{s0}\cdot\sin^2\phi_1+\Delta y_{s1}\cdot\cos^2\phi_1\right) \tag{4.18}$$

From equations (4.17) and (4.18) we find that:

$$\Delta L_p(\Delta t_2) = \Delta L_N(\Delta t_2)\cdot \tan \phi_1 + \frac{1}{\cos \phi_1} [v_y \Delta t_2 - \Delta y_{s1}] \tag{4.19}$$

Then we have:

$$R_1 + \Delta L_p(\Delta t_2) = \frac{R_0 \cos\phi_0 + \Delta L_N(\Delta t_2)\sin\phi_1 + v_y(\Delta t_1 + \Delta t_2) - (\Delta y_{s0} + \Delta y_{s1})}{\cos \phi_1} \tag{4.20}$$

Inserting for $(R_1 + \Delta Lp(\Delta t_2))$ from equation (4.20) in equation (4.3) gives:

$$\frac{\Delta L_N(\Delta t_2)\cos\phi_1}{\tan(\phi_2 - \phi_1)} = R_0 \cos \phi_0 + \Delta L_N(\Delta t_2)\sin \phi_1 + v_y(\Delta t_1 + \Delta t_2) - (\Delta y_{s0} + \Delta y_{s1}) \tag{4.21}$$

Since $\Delta L_N(\Delta t_2)$ given by equation (4.17) is not a function of v_y, and since:

$$\Delta t_1 + \Delta t_2 = t_2 - t_0 \tag{4.22}$$

equation (4.21) can be solved for v_y. The result is:

$$v_y = \frac{1}{t_2 - t_0}\left[\frac{\Delta L_N(\Delta t_2)\cos \phi_2}{\sin(\phi_2 - \phi_1)} - R_0 \cos \phi_0 + \Delta y_{s0} + \Delta y_{s1}\right] \tag{4.23}$$

Inserting for $\Delta L_N(\Delta t_2)$ from equation (4.17) gives our final result for v_y:

$$v_y = \frac{1}{t_2 - t_0}\left[R_0\left\{\frac{\Delta t_2}{\Delta t_1}\ \frac{\sin(\phi_1 - \phi_0)\cos \phi_2}{\sin(\phi_2 - \phi_1)} - \cos \phi_0\right\} + \Delta y_{s0} + \Delta y_{s1} + \frac{\cos \phi_2}{\sin(\phi_2 - \phi_1)}\left[\cos \phi_1\left(\frac{\Delta t_2}{\Delta t_1}\Delta x_{s0} - \Delta x_{s1}\right) - \sin\phi_1\left(\frac{\Delta t_2}{\Delta t_1}\Delta y_{s0} - y_{s1}\right)\right]\right] \tag{4.24}$$

Lastly, inserting for v_y from equation (4.24) in equation (4.15), gives our final result for v_x:

$$v_x = \frac{1}{t_2-t_0}\left[R_0\left\{\frac{\Delta t_2}{\Delta t_1}\frac{\sin(\phi_1-\phi_0)\sin\phi_2}{\sin(\phi_2-\phi_1)} - \sin\phi_0\right\} + \Delta x_{s0} + \Delta x_{s1} + \frac{\sin\phi_2}{\sin(\phi_2-\phi_1)}\left[\cos\phi_1\left(\frac{\Delta t_2}{\Delta t_1}\Delta x_{s0}-\Delta x_{s1}\right) - \sin\phi_1\left(\frac{\Delta t_2}{\Delta t_1}\Delta y_{s0}-\Delta y_{s1}\right)\right]\right] \tag{4.25}$$

Equations (4.24) and (4.25) are the main results in this section. We are interested in their form for a couple of special cases:

<u>Case 1</u>:

$$\Delta t_1 = \Delta t_2 = \Delta t \tag{4.26}$$

Observer non-maneuvering, i.e.:

$$\Delta x_{s0} = \Delta x_{s1} = v_{sx}\cdot\Delta t \tag{4.27}$$

$$\Delta y_{s0} = \Delta y_{s1} = v_{sy}\cdot\Delta t \tag{4.28}$$

$$v_x = v_{sx} + \frac{R_0}{2\cdot\Delta t}\left\{\frac{\sin(\phi_1-\phi_0)\sin\phi_2}{\sin(\phi_2-\phi_1)} - \sin\phi_0\right\} \tag{4.29}$$

$$v_y = v_{sy} + \frac{R_0}{2\cdot\Delta t}\left\{\frac{\sin(\phi_1-\phi_0)\cos\phi_2}{\sin(\phi_2-\phi_1)} - \cos\phi_0\right\} \tag{4.30}$$

As we can see from equation (4.29) and (4.30), the target velocities that can be determined on the basis of <u>any</u> three consecutive bearings from a non-maneuvering target, will depend on the range R_0 (can be transformed to yield R_1 by solving equation (4.16) for R_0 after inserting for v_y. Similarily, the range dependence can also be transformed to R_2).

It is important to realize that the same dependence on range exists when the target velocities are calculated by the Kalman filter. This is the range/velocity ambiguity in a nut-shell!

Case 2:

$$\Delta t_1 = \Delta t_2 = \Delta t$$

Observer not moving at all, i.e.:

$$\Delta x_{s0} = \Delta x_{s1} = 0 \tag{4.31}$$

$$\Delta y_{s0} = \Delta y_{s1} = 0 \tag{4.32}$$

$$v_x = \frac{R_0}{2 \cdot \Delta t} \left\{ \frac{\sin(\phi_1 - \phi_0) \sin \phi_2}{\sin(\phi_2 - \phi_1)} - \sin \phi_0 \right\} \tag{4.33}$$

$$v_y = \frac{R_0}{2\Delta t} \left\{ \frac{\sin(\phi_1 - \phi_0) \cos \phi_2}{\sin(\phi_2 - \phi_1)} - \cos \phi_0 \right\} \tag{4.34}$$

In this case, it is possible to determine the target course independent of range. We have:

$$C_T = \tan^{-1}\left(\frac{v_x}{v_y}\right) = \tan^{-1}\left(\frac{\sin(\phi_1-\phi_0)\sin \phi_2 - \sin\ (\phi_2-\phi_1)\sin\phi_0}{\sin(\phi_1-\phi_0)\cos\phi_2 - \sin(\phi_2-\phi_1)\cos\phi_0}\right) \tag{4.35}$$

The target velocity, however, will still depend on the range.

These two special cases also can be used to give guidelines for an intelligent observer maneuver strategy: Start tracking while the observer stays put, and determine the target course. When the observer starts moving, preserve the determined target course in the Kalman filter (by keeping σ_c low, may be artifically). Then the filter will quickly adopt the correct range. An indication of this can be found from equations (4.29) and (4.30). We can write:

$$\tan C_T = \frac{v_x}{v_y} = \frac{v_{sx} + R \cdot K_1}{v_{sy} + R \cdot K_2} \tag{4.36}$$

where C_T is assumed known, and K_1 and K_2 are constants, independent of range. Equation (4.36) can be solved for R, to give:

$$R = \frac{v_{sx} - v_{sy} \cdot \tan C_T}{K_2 \cdot \tan C_T - K_1} \tag{4.37}$$

4.4 Calculation of Initial Range

By inclusion of a 4th bearing observation, the results of the previous section can be extended to include the determination of the initial range, provided the observer is maneuvering during the observation period.

When a 4th bearing is included, the following equation can be included:

$$\frac{\Delta L_N(\Delta t_3)}{R_2 + \Delta L_p(\Delta t_3)} = \tan(\phi_3 - \phi_2) \tag{4.38}$$

Here, R_2 is given by:

$$R_2 = \sqrt{(R_1 + \Delta L_p(\Delta t_2))^2 + \left(\Delta L_N(\Delta t_2)\right)^2} \tag{4.39}$$

Since, from equation (4.3):

$$\Delta L_N(\Delta t_2) = (R_1+\Delta L_p(\Delta t_2))\tan(\phi_2-\phi_1) \tag{4.40}$$

equation (4.39) can be written:

$$R_2 = \left[R_1 + \Delta L_p(\Delta t_2)\right] / \cos(\phi_2-\phi_1) \tag{4.41}$$

Now, by use of equation (4.20) we get:

$$R_2 = \frac{R_0\cos\phi_0 + \Delta L_N(\Delta t_2)\sin\phi_1 + v_y(\Delta t_1+\Delta t_2)-(\Delta y_{s0}+\Delta y_{s1})}{\cos\phi_1 \cos(\phi_2-\phi_1)} \tag{4.42}$$

The next step is to insert for $\Delta L_N(\Delta t_2)$ from equation (4.17) and for v_y from equation (4.24) in equation (4.42). After some manipulations, we get the following result:

$$R_2 = \frac{\frac{\Delta t_2}{\Delta t_1} R_0\sin(\phi_1-\phi_0) + \cos\phi_1\left(\frac{\Delta t_2}{\Delta t_1}\Delta x_{s0} - \Delta x_{s1}\right) - \sin\phi_1\left(\frac{\Delta t_2}{\Delta t_1}\Delta y_{s0} - \Delta y_{s1}\right)}{\sin(\phi_2-\phi_1)} \tag{4.43}$$

Next, the two other variables in equation (4.38), $\Delta L_N(\Delta t_3)$ and $\Delta L_p(\Delta t_3)$ have to be calculated. We have:

$$\Delta L_N(\Delta t_3) = \cos\phi_2[v_x\Delta t_3 - \Delta x_{s2}] - \sin\phi_2[v_y\Delta t_3-\Delta y_{s2}] \tag{4.44}$$

$$\Delta L_p(\Delta t_3) = \sin\phi_2[v_x\Delta t_3 - \Delta x_{s2}] + \cos\phi_2[v_y\Delta t_3-\Delta y_{s2}] \tag{4.45}$$

Further:

$$\Delta L_p(\Delta t_3) = \Delta L_N(\Delta t_3) \cdot \tan\phi_2 + \frac{1}{\cos\phi_2}[v_y\Delta t_3-\Delta y_{s2}] \tag{4.46}$$

Now, inserting for $\Delta L_p(\Delta t_3)$ from equation (4.46) in equation (4.38) gives, after some rearrangements:

$$\Delta L_N(\Delta t_3) \cdot \frac{\cos\phi_3}{\sin(\phi_3-\phi_2)} = R_2 \cos\phi_2 + v_y\Delta t_3 - \Delta y_{s2} \tag{4.47}$$

Next, we have to develop $\Delta L_N(\Delta t_3)$ further. Inserting for v_x and v_y from equations (4.25) a. (4.24) in equation (4.44) gives, after some calculation:

$$\Delta L_N(\Delta t_3) = \frac{R_0\Delta t_3}{t_2-t_0}\sin(\phi_2-\phi_0) + \cos\phi_2\left[\frac{\Delta t_3}{t_2-t_0}(\Delta x_{s0} + \Delta x_{s1})-\Delta x_{s2}\right]$$
$$- \sin\phi_2\left[\frac{\Delta t_3}{t_2-t_0}(\Delta y_{s0}+\Delta y_{s1})-\Delta y_{s2}\right] \tag{4.48}$$

Lastly, by inserting for $\Delta L_N(\Delta t_3)$ from equation (4.48), for R_2 from equation (4.43) and for v_y from equation (4.24), equation (4.47) can be solved for R_0. The result is:

$$R_0 = \frac{\sin(\phi_2-\phi_1)\cdot\Delta t_1\{\sin\phi_3(\Delta t_3(\Delta y_{s0}+ y_{s1})-(t_2-t_0)\Delta y_{s2})-\cos\phi_3(\Delta t_3(\Delta x_{s0}+}{\Delta t_1\cdot\Delta t_3\cdot\sin(\phi_3-\phi_0)\sin(\phi_2-\phi_1) -}$$

$$\frac{\Delta x_{s1})-(t_2-t_0)\Delta x_{s2})\} + \sin(\phi_3-\phi_2)(t_3-t_0)\{\cos\phi_1(\Delta t_2\Delta x_{s0}-\Delta t_1\Delta x_{s1})-\sin\phi_1(\Delta t_2\Delta y_{s0}-\Delta t_1\Delta y_{s1}}{\Delta t_2(t_3-t_0)\sin(\phi_1-\phi_0)\sin(\phi_3-\phi_2)} \tag{4.49}$$

Equation (4.49) is the final result in this section. An important special case is obtained when:

$$\Delta t_1 = \Delta t_2 = \Delta t_3 = \Delta t$$
$$t_2-t_0 = 2\Delta t \tag{4.50}$$
$$t_3-t_0 = 3\Delta t$$

Then, equation (4.49) can be written:

$$R_0 = \frac{\sin(\phi_2-\phi_1)\{\sin\phi_3(\Delta y_{s0}+\Delta y_{s1}-2\Delta y_{s2})-\cos\phi_3(\Delta x_{s0}+\Delta x_{s1}-2\Delta x_{s2})\} + 3\sin(\phi_3-\phi_2)\cdot\{\cos\phi_1(\Delta x_{s0}-\Delta x_{s1})-\sin\phi_1(\Delta y_{s0}-\Delta y_{s1})\}}{\sin(\phi_3-\phi_0)\sin(\phi_2-\phi_1)-3\sin(\phi_1-\phi_0)\sin(\phi_3-\phi_2)} \tag{4.51}$$

If the observer is not maneuvering, we have:

$$\left.\begin{aligned} &\Delta x_{s0} = \Delta x_{s1} = \Delta x_{s2} \\ \text{and} \\ &\Delta y_{s0} = \Delta y_{s1} = \Delta y_{s2} \end{aligned}\right\} \tag{4.52}$$

From equation (4.51) it is then easy to see that $R_0=0$, meaning that range can not be observed with a nonmeaneuvering observer.

When range R_0 is calculated from equation (4.49) or (4.51), equations (4.24) and (4.25) can be used with this range R_0 to calculate the target velocity components.

Appendix F gives a numerical example on the use of the results in section 4.3 and 4.4.

4.5 State vector initialization.

Due to obvious reasons, the single Kalman filter case and the parallel filter case have to be treated separately under this heading.

4.5.1 Single Kalman filter case.

The results in Section 4.3 and 4.4 are based on the assumption of noisefree bearings. They can, however, be used directly as initial data, with the associated initial values for the covariance matrix as given in section 4.6.3 and in appendices G and H.

In the following, however, two different methods will be suggested in order to improve the initial values of velocity and range. The approaches will depend on whether the observer is maneuvering or not over the time period [0, NT], which we will call the initialization period.

We assume that the following data are available:

1. A sequence of bearing observations, $Z_N = \{\phi_0, \phi_1 \ldots \phi_N\}$
2. The observers position increments, $\Delta x_{s_{N-1}} = \{\Delta x_{s0}, \Delta x_{s1} \ldots \Delta x_{s_{n-1}}\}$ and $\Delta y_{s_{N-1}} = \{\Delta y_{s0}, \Delta y_{s1}, \ldots, \Delta y_{s_{N-1}}\}$.

Case 1, Observer Maneuvering

Step 1: Calculate: $R_i = f(\phi_i, \phi_{i+1}, \phi_{i+2}, \phi_{i+3})$, $i=0,1,\ldots,N-3$. The function f is defined by equation (4.49).

Step 2: Calculate: $v_{xi} = f_1(R_i, \phi_i, \phi_{2+1}, \phi_{i+2})$, $i=0,1,\ldots,N-3$.

$$v_{yi} = f_2(R_i, \phi_i, \phi_{i+1}, \phi_{i+2}) \quad i=0,1,\ldots,N-3.$$

where the functions f_1 and f_2 are defined by equations (4.25) and (4.24).

Step 3: Calculate the initial velocity elements as:

$$\bar{v}_{x0} = \frac{1}{N-2} \sum_{i=0}^{N-3} v_{xi} \tag{4.53}$$

$$\bar{v}_{y0} = \frac{1}{N-2} \sum_{i=0}^{N-3} v_{yi} \tag{4.54}$$

Step 4: Calculate the target course as

$$C_T = \tan^{-1}\left(\frac{\bar{v}_{x0}}{v_{y0}}\right) \tag{4.55}$$

<u>Step 5:</u> Calculate initial range, R_0, from equation (4.37).

<u>Step 6:</u> Initial bearing is given as ϕ_0.

<u>Step 7:</u> Initial position elements are calculated as:

$$x_0 = x_{s0} + R_0 \sin \phi_0 \tag{4.56}$$

$$y_0 = y_{s0} + R_0 \cos \phi_0 \tag{4.57}$$

<u>Step 8:</u> Starting with $\underline{x}_0 = [x_0\ y_0\ \bar{v}_{x0}\ \bar{v}_{y0}]^T$, process the bearings $\{\phi_1, \phi_2, \ldots, \phi_N\}$ through the Kalman filter, to give the resulting state vector at $t = t_N$, where the track is offically started.

<u>Case 2, Observer not maneuvering</u>

In this case, equation (4.49) is worthless, and no range information can be subtracted directly from the bearings. In this case, the following steps are proposed:

<u>Step 1:</u> Select R_0 as the sensor's maximum range. (If possible, take environmental conditions into account).

<u>Step 2:</u> Calculate: $v_{xi} = f_1(R_i, \phi_i, \phi_{i+1}, \phi_{i+2})$

$$v_{yi} = f_2(R_i, \phi_i, \phi_{i+1}, \phi_{i+2})$$

$$R_{i+1} = f_3(R_i, v_{yi}, \phi_i, \phi_{i+1})$$

$i = 0,1,2,\ldots,N-3$. The functions f_1 and f_2 are given by equations (4.25) and (4.24), while the function f_3 can be found by an extension of equation (4.16), as:

$$R_{i+1} = \frac{R_i \cdot \cos \phi_i + (v_{yi}\Delta t_i - \Delta x_{si})}{\cos \phi_{i+1}} \qquad (4.58)$$

<u>Step 3</u>: Calculate:

$$v_x = \frac{1}{N-2} \sum_{i=0}^{N-3} v_{xi} \qquad (4.59)$$

$$v_y = \frac{1}{N-2} \sum_{i=0}^{N-3} v_{yi} \qquad (4.60)$$

<u>Step 4</u>: If target identification and classification has been performed the knowledge of this target class's maximum velocity, v_{MAX}, will presumably be known. Otherwise, some value of v_{MAX} can always be specified for submarines, surface ships, etc. Calculate the initial velocity components as:

$$\bar{v}_{x0} = \min\left(\frac{v_{MAX}}{\sqrt{v_x^2 + v_y^2}} \cdot v_x, v_x\right) \qquad (4.61)$$

$$\bar{v}_{y0} = \min\left(\frac{v_{MAX}}{\sqrt{v_x^2 + v_y^2}} \cdot v_y, v_y\right) \qquad (4.62)$$

<u>Steps 5-9</u>: Same as Case 1, Steps 4-8.

4.5.2 Parallel Filter Case

For this approach, when we have M parallel filters to initialize, a slightly different initialization scheme has to be suggested.

We assume that the same data are available as in Section 4.5.1, and we will have the same two cases, depending on the observer's maneuvering scheme:

Case 1, Observer Maneuvering.

Steps 1-5: Same as in Section 4.5.1, case 1.

Step 6: With a previously defined ΔR_1, calculate R_{0i}, i=1,2,...,M, as:

$$R_{0i} = R_0 - \frac{M}{2} \cdot \Delta R_1 + (i - \frac{1}{2})\Delta R_1 \tag{4.63}$$

Step 7: Calculate the velocity components $v_{x_{0i}}$ and $v_{y_{0i}}$ from equations (4.25) and (4.24), i=1,2,...,M.

Step 8: Initial bearing for each of the M filters is given as $\phi_{0i} = \phi_0$, i=1,2,...,M.

Step 9: Initial position data are calculated as:

$$x_{0i} = x_{s0} + R_{0i} \cdot \sin \phi_0 \tag{4.64}$$

$$y_{0i} = y_{s0} + R_{0i} \cos \phi_0 \tag{4.65}$$

i=1,2,...,M.

Step 10: Same as Section 4.5.1, case 1, step 8 for each filter.

Case 2, Observer not maneuvering

Steps 1-6: Same steps as in section 4.5.1, case 2, resulting in R_0 at the end of step 6.

Step 7: With a previously defined $\Delta R_2 > \Delta R_1$, calculate R_{0i}, $i=1,2,\ldots,M$, as:

$$R_{0i} = R_0 - i \cdot \Delta R_2 \tag{4.66}$$

(R_0 is assumed to be near the maximum range for the bearing sensor, and any $R_{0i} > R_0$ should not be necessary).

Steps 8-11: Same steps as Case 1 above, steps 7-10.

4.6 Covariance Matrix Initialization

The next subject to be addressed, is the selection/calculation of initial values for the covariance matrix, P_0. We will only derive the initial covariance matrix for the Cartesian system model. (Necessary transformation to the Polar coordinate system case can be done through equation (3.48)).

Since the philosophy behind the parallel filter approach is totally different from the single Kalman filter approach with respect to assumed initial accuracy in range, these two cases have to be treated separately also under this heading.

4.6.1 Single Kalman Filter Case

Three different approaches to covariance matrix initialization will be outlined under this heading.

4.6.1.1 Aidala's Approach

Aidala [11] has treated this subject very thoroughly, however, his model of the target motion analysis problem has no process noise.

Consequently, his time update equation for the covariance matrix has the form:

$$P_{k+1,k} = \phi(T) \cdot P_{k,k} \phi(T)^T \tag{4.67}$$

i.e., no increase in uncertainty with time, since the term $\theta(T) V_k \theta(T)^T$ from equation (3.18) is lacking. Since his conclusions depends entirely on this unrealistic assumption, his results are not applicable directly.

Aidala proposes to select

$$P_0 = \sigma_0^2 \cdot I \tag{4.68}$$

and shows this selection's superiority in terms of covariance matrix stability over the selection

$$P_0 = \begin{bmatrix} \sigma_r^2 & & & 0 \\ & \sigma_r^2 & & \\ & & \sigma_v^2 & \\ 0 & & & \sigma_v^2 \end{bmatrix} \tag{4.69}$$

By a pseudolinear formulation of the tracking problem, obtained through a nonlinear transformation, Aidala transforms the nonlinear observation equation to a linear equation, with the nonlinearities embedded in the observation noise (see also [7]. The measurement standard deviation for this model turns out to be:

$$\sigma_{z_k} = [R_{k,k-1}/\sigma_0] \cdot \sqrt{W_k} \tag{4.70}$$

and this is the only term containing R and σ_0 in the covariance equations for the pseudo-linear filter.

The covariance matrix of the pseudolinear filter, $\hat{P}_{k,k}$, is related to the covariance matrix of the ori/inal filter, $P_{k,k}$, in the following way, if equation (4.68) is used for initialization:

$$P_{k,k} = \sigma_0^2 \; \hat{\ }_{:,k} \tag{4.71}$$

By selecting $\sigma_0 = R_0$, equation (4.70) can be approximated as:

$$\sigma_{z_k} = \sqrt{W_k} \tag{4.72}$$

and the covariance equations for the pseudolinear filter is completely independent of range and $\sigma_0 (\hat{P}_{0,0} = I)$.

However, the reason why this form is possible, is the form of the covariance time update equation given by equation (4.67). If equation (3.18) was used, the initial value of the variance, σ_0^2, can not be collapsed into the measurement variance term, as given by equation (4.70).

Now, is it feasible to adapt Aidala's results for our model, where time updating of the covariance matrix is performed according to equation (3.18)?

This question is not possible to answer without performing simulations.

For the case with a nonmaneuvering observer during the initialization phase, it is therefore suggested to incorporate the selection of the initial covariance matrix

$$P_0 = R_0^2 \cdot I \tag{4.73}$$

as one possibility to be explored.

4.6.1.2 General Approach

The most commonly used initial covariance matrix is probably of the form:

$$P_{0,-1} = P_0 = \begin{bmatrix} p_{11} & p_{12} & 0 & 0 \\ p_{21} & p_{22} & 0 & 0 \\ 0 & 0 & p_{33} & p_{34} \\ 0 & 0 & p_{43} & p_{44} \end{bmatrix} \tag{4.74}$$

where:

$$p_{11} = \cos^2\phi_0 \cdot (R_0 \cdot \sigma_{\phi_0})^2 + \sin^2\phi_0 \cdot \sigma_{R_0}^2 \tag{4.75}$$

$$p_{12} = p_{21} = \cos\phi_0 \sin\phi_0 \cdot [\sigma_{R_0}^2 - (R_0\sigma_{\phi_0})^2] \tag{4.76}$$

$$p_{22} = \sin^2\phi_0 \cdot (R_0\sigma_{\phi_0})^2 + \cos^2\phi_0 \cdot \sigma_{R_0}^2 \tag{4.77}$$

$$p_{33} = \cos^2 c_0 \cdot (v_0 \cdot \sigma_{c_0})^2 + \sin^2 c_0 \cdot \sigma_{v_0}^2 \tag{4.78}$$

$$p_{34} = p_{43} = \sin c_0 \cdot \cos c_0 [\sigma_{v_0}^2 - (v_0 \cdot \sigma_{c_0})^2] \tag{4.79}$$

$$p_{44} = \sin^2 c_0 (v_0\sigma_{c_0})^2 + \cos^2 c_0 \cdot \sigma_{v_0}^2 \tag{4.80}$$

The usual (ad hoc) selection of parameter values in equations (4.75) - (4.80) is the following:

ϕ_0 = the observed bearing at time t_0

σ_{ϕ_0} = the sensor's standard deviation

R_0 = the sensor's max range.

$\sigma_{R_0} = \frac{1}{2}$ (the sensors max range)

$c_0 = -\phi_0$

v_0, σ_{c_0} and σ_{v_0} : completely picked out of the air. As an example:

v_0 = 10 m/sec

$\sigma_{c_0} = 30°$

σ_{v_0} = 20 m/sec.

4.6.1.3 Suggested Approach

If the state initialization methods given in section 4.3 and 4.4 are used, a better initial covariance matrix can be obtained.

Like in the state vector initialization case, the approach will depend on the observers maneuvering scheme:

Case 1, Observer Maneuvering.

Step 1: Range can now be calculated from equation (4.49). Calculate the initial variance for range, $\sigma^2_{R_0}$, through the approach given in Appendix G.

Step 2: The velocity elements can be calculated from equations (4.24) and (4.25). Calculate the velocity elements of the covariance matrix through the approach given in Appendix H.

When the "smoothing" approach given in section 4.5.1 is used, the resulting initial values of the state vector should be better than the resulting initial covariance matrix from this approach will indicate. However, since we are interested in an initially "open" filter (a filter that responds to the measurements), we suggests not to take this into account. If the actual accuracy obtained through the methods given in section 4.5.1 turns out to be much higher than the resulting covariance from this approach, it will, however, be worthwhile to take the "smoothing" effect into account also for initial covariance calculations.

4.6.2 Parallel Filter Case

With M filters running in parallel, we are interested in each filter being as "stiff" as possible in range, since it is likely that one of the filters have an initial range close to the correct one, as will subsequently become apparent through an observer maneuvre (see section 3.7 and 4.5.2).

For the parallel filter approach, the only difference between the two cases: observer maneuvering/not maneuvering, is the size ΔR between the initial values of range for each filter (see section 4.5.2, ΔR_1 or ΔR_2).

The following approach is suggested:

Step 1: Select $\sigma_{R_0} = \frac{\Delta R}{2}$.

Step 2: Select σ_{ϕ_0} = the sensors standard deviation.

Step 3: Given the initial range R_{0i}, $i=1,2,\ldots,M$, calculate the position covariance elements for each filter from equations (4.75)-(4.77).

Step 4: Calculate the velocity elements of the covariance matrix for each filter through the approach given in Appendix H.

5. MANEUVRE DETECTION AND HANDLING

Up to this point we have assumed constant course and speed for the target. This assumption will now be removed, and we will allow our target to make abrupt changes in course and/or speed, of random sizes, and occuring at random time instants.

Several different approaches to the maneuvering target tracking problem have been proposed in the literature. We will in the following give a short survey of some of the most important approaches.

Jazwinski's [25] limited memory filtering approach is probably the simplest approach. By preventing the covariance matrix elements from decaying below certain thresholds, resulting in filter gains above certain values, the target state vector dependence on the latest observations are increased. However, the tracking performance of this approach during nonmaneuvering periods of the target will decrease due to higher dependence on the observation noise.

The natural solution to this problem is to model the target under the nonmaneuvering hypothesis, and in addition to introduce some adaptive maneuvre detection scheme which can step in and give the filter limited memory for a short period of time, after a maneuvre is detected [4], [20].

A further extension to this approach is suggested by Willsky and Jones [26], [27]. They suggest the possibility of simultaneously with the detection of an abrupt system change, to estimate the size of the change and to perform a state variable correction directly, in addition to give the filter limited memory temporarily. Similar approaches are outlined in [9] and [10].

Another avenue along which many researchers have been working, is to model the target maneuvres as a semi-Markov process, whereby N possible acceleration inputs are selected according to some a priori probabilities. In its pure form, this approach requires an infinitely growing bank of parallel filters, N initially, N^2 for the second measurements, etc. Different approaches to reduce this described growth are proposed in the literature, in order to get practical, realizable filters [14], [10], [21], [22], [28], [29].

Another interesting approach was proposed by Tenney et al. [24]. Two extended Kalman filters are operating in parallel, one with a large artificial system noise covariance term added to give the filter limited memory and thereby allowing it to track fast maneuvres, and the other filter with small artificial noise, making this filter restricted to tracking of constant course/speed trajectories. Maneuvre detection is performed by comparing the behaviour of the two filters.

None of the approaches to maneuvering target tracking resumed above, deal with a tracking scheme based on bearing only information. Due to the low observability of the system, maneuvre detection under these circumstances are more difficult. Most of the papers above claim for high observability in order to make efficient maneuvering target trackers.

Another important fact to realize is the possibility to separate the maneuvering target tracking problem into two subproblems:

1. The maneuvre detection problem.

2. The maneuvre handling problem (i.e., When the maneuvre is detected, what actions should be taken to allow the tracker to adapt to the new target course/speed).

This separation is imbedded in the approaches given in [4], [9], [24], [26] [27], and will also be used in the approach proposed in this report.

The lacking observability during time-periods when the observer is nonmaneuvering, results, as we have seen in Chapter 4, in the range/velocity ambiguity. Unless special preventing actions are taken, the Kalman filter's reaction to a target maneuvre can be a range jump as well as a course/velocity jump. The obvious preventing action is to keep the range variance σ_R^2 low, forcing the filter to adapt course/velocity as a maneuvre detection reaction, and leave the range to target unchanged. This seems, however, only feasible to do if we have arrived at a stable target track with correct range prior to the maneuvre. If the target maneuvre takes place while the tracker still is in the initialization phase, with a poor linearization trajectory in range, our wish is to keep a high range variance σ_R^2 in the Kalman filter, so that the filter easily can arrive at correct range if/when the observer performs a maneuvre, and range becomes observable.

These conflicting preferences on σ_R^2 are among the reasons why maneuvering target tracking is harder to solve for the bearing only measurement case than for cases with complete observability.

One possible way to resolve this conflict, is to make use of our knowledge of the observers position history and future maneuvre intentions. If, for the global iterated filter case, the observer has not performed maneuvres during the iteration interval, and generally for all the filter approaches, if the observer is not going to start maneuvering in the near future, it is no point in keeping σ_R^2 high, since range can't be observed from the observations in any case. However, we will get a possible conflict if the observer and the target are maneuvering at the same time. The solution to this problem has to be decided upon through simulations.

A proposed approach to maneuvre detection and handling will be given in the following.

5.1 Maneuvre Detection

The most powerful and best theoretical fundamented approach to maneuvre detection seems to be the generalized likelihood ratio best described in [30]. This approach has been used by Willsky and Jones [26], [27], by Tenney et al. [4], and is also suggested by Maybeck [23].

Following the approach given in these references, two hypothesis on the form of the innovation signal can be assumed:

H_0: $\varepsilon_k = \nu_{1k}$ (No maneuvre)

H_1: $\varepsilon_k = m_k + \nu_{1k}$ (A maneuvre has occurred.)

where ν_{1k} is a zero mean white sequence with variance:

$$\sigma_{\nu_{1k}}^2 = H_k \cdot P_{k,k-1} H_k^T + W_k \tag{5.1}$$

If we restrict our attention to a "data window" containing the N most recent observations, our generalized likelihood ratio test can be given by:

$$L_k = c_k - \sum_{i=k-L}^{k} \frac{\varepsilon_i^2}{\sigma^2_{\nu_{1i}}} \underset{H_1}{\overset{H_0}{\gtrless}} \gamma \tag{5.2}$$

where c_k, L and γ are design values which has to be decided upon through simulations. c_k is a (possible) varying term independent of the observed residual values. If we define:

$$\gamma_{1k} = c_k - \gamma \tag{5.3}$$

the likelihood ratio test can be given by

$$\sum_{i=k-L}^{k} \frac{\varepsilon_i^2}{\sigma^2_{\nu_{1i}}} \underset{H_0}{\overset{H_1}{\gtrless}} \gamma_{1k} \tag{5.4}$$

The reasons why we incorporate the possibility of having a varying threshold γ_{1k} are the following:

1. We want to inhibit maneuvre detection during initialization phase.

2. We want to inhibit successive maneuvre detections during the same maneuvre.

3. We want to inhibit maneuvre detections during maneuvering phases of the observer, if we have reason to believe that our linearization trajectory prior to the observer maneuvre is poor.

The meaning of equation (5.4) is obvious: The actual variance of the innovation signal is compared with its expected value over the most recent L samples. If ε_k^2 becomes consistently larger than predicted over the selected "time window", a target maneuvre is detected.

A few guidelines can be given concerning the selection of the design parameters L, c_k and γ (alternatively L and γ_{1k}):

L has to be selected as a compromise between fast detection and the false detection rate. The longer time window the slower maneuvre detection and the less probability for false maneuvre detection. The shorter time window, the faster maneuvre detection, but at the same time, the higher probability for false detections.

γ can be decided upon by looking at the case with stationary circumstances: Tracking of a target going with constant course and speed, where our linearization trajectory is approximately correct. With $c_k = 0$ and the process noise covariance matrix V_k, the observation noise variance W_k and the "time window" (decided by L) given, γ can be selected to give a false alarm probability close to zero.

c_k has to depend on a number of parameters, and may have different size and time dependence, depending on the state of the filter. We may suggest the following structure of c_k:

$$c_k = \begin{cases} K\, e^{-\alpha(k-k_1)T} & k > k \\ 0 & k \leq k_1 \end{cases} \tag{5.5}$$

where T is the sample time, and k_1 is the sample when a certain event takes place, like:

1. Initialization
2. Maneuvre detection
3. Starting of an observer maneuvre

The parameters K and α are event-dependent parameters which has to be selected through simulations in order to give acceptable false alarm probability in the transient-period following the actual event. These parameters will also depend on the "time window" LT, on V_k, W_k, P_0 and on possible direct manipulation on the covariance matrix following a detected maneuvre.

The maneuvre detection approach described in this section should be suitable for all the filtering approaches given in Chapter 3. Whether it is the best approach to maneuvre detection, independent of the filtering approach, is unknown.

For the iterated filtering approaches given in Chapter 3.4 to 3.6, we would suggest to include an the simulation study an investigation on which value of the innovation signal that should be used in equation (5.4), the first or the last iteration value.

For the parallel filter case, we propose to run the maneuvre detection algorithm only on one filter, namely the filter which, at each time instant kT, has the highest probability function $p(\hat{R}_{ki} = R_k/z_k)$.

5.2 Actions Following a Detected Maneuvre

As described in the beginning of this Chapter, the most appropriate action to take when a maneuvre is detected, is to give the filtering algorithm limited memory temporarily, allowing the filter to adapt to the new target course and speed more easily.

One important fact to realize, however, is that there is a certain time delay between the beginning of a maneuvre and its detection. This time delay, say Δt, depends on a number of factors, however, some nominal value of Δt should be possible to obtain through simulations.

If we don't take this time delay into account, and impose limited memory on the filtering algorithm only from the time instant kT, when the maneuvre is detected, the observations taken during the intermediate period from the occurrence of the maneuvre and its detection, will not be utilized optimally. The result will be an accumulated range error, that can not be driven to zero unless the observer performs a maneuvre <u>after</u> the target maneuvre.

In order to reduce this range error, even for the case of a non-maneuvering observer subsequent to a target maneuvre, the following steps are suggested:

1. When a maneuvre is detected, fetch a stored version of the state vector and the covariance matrix valid at time ($kT-\Delta t$) from the computer memory.
2. Give this time version of the filter limited memory by increasing the velocity elements of the covariance matrix.
3. Re-integrate all the observations taken in the time interval $[kT-\Delta t, kT]$.

The result of this iteration will be a discrete jump to a more correct position and velocity for the state vector at time kT, the same time instant when the target maneuvre was detected. The effect of this approach is thus the same as achieved by Willsky and Jones [26], [27], and estimate of the size of system change, and a direct correction of the state vector. The methods are, however, different.

The prize that has to be paid for this achievement, however, is substantial. The following time sequences has to be stored in the computer memory:

Observation sequence:

$$Z_k = \{z_{kT-\Delta t}, z_{(k+1)T-\Delta t} \cdots z_{kT}\} \tag{5.6}$$

Observer x, y-position:

$$X_s = \{x_{s(kT-\Delta t)}, \ldots, x_{s_{kT}}\} \tag{5.7}$$

$$Y_s = \{y_{s(kT-\Delta t)}, \ldots, y_{s_{kT}}\} \tag{5.8}$$

Target state vector

$$X_k = \{\underline{x}_{kT-\Delta t}, \ldots, \hat{\underline{x}}_{kT}\} \tag{5.9}$$

covariance matrix

$$\underline{P}_k = \{P_{kT-\Delta t}, \ldots, P_{kT}\} \tag{5.10}$$

(Since we don't know when a maneuvre detection might occur, we have to store the sequences given in eq. (5.9) and (5.10), even if the only values of interest to us at the time of maneuvre detection are $\hat{\underline{x}}_{kT-\Delta t}$).

It is interesting to realize, that in the case with global iterated filters, the sequences given by equation (5.6)-(5.8) and (5.10) are already stored, if $MT \geq \Delta t$ (see section 3.6). If we adopt the global iterated approach, it is not necessary to store the state vector sequence given by equation (5.9). The state vector at time $kT-\Delta t$ can then be found by time-backdating. This approach should be investigated through simulations.

5.2.1 Imposing Limited Memory on the Filter

The course and velocity variance can be calculated from the velocity elements of the covariance matrix (both in the Cartesian and the

Polar coordinate system case) through the following equations:

$$\sigma_v^2 = p_{33}\cdot\sin^2 c - (p_{34}+p_{43})\sin c \cos c + p_{44}\cos^2 c \tag{5.11}$$

$$\sigma_c^2 = \frac{1}{v^2}\,[p_{33}\cos^2 c + (p_{34}+p_{43})\sin c \cos c + p_{44}\sin^2 c] \tag{5.12}$$

Following a target maneuvre detection, we now want to increase σ_c by $\Delta\sigma_c$ and σ_v by $\Delta\sigma_v$. Then, if we define:

$$\delta v = (\sigma_v+\Delta\sigma_v)^2-\sigma_v^2 \tag{5.13}$$

$$\delta c = (\sigma_c+\Delta\sigma_c)^2 - \sigma_c^2 \tag{5.14}$$

the result on the velocity elements of the covariance matrix will be:

$$p_{33} \leftarrow p_{33} + v^2\cos^2 c \cdot \delta c + \sin^2 c \cdot \delta v \tag{5.15}$$

$$p_{34} \leftarrow p_{34} + \sin c \cos c[\delta v - v^2\delta c] \tag{5.16}$$

$$p_{43} \leftarrow p_{43} + \sin c \cos c[\delta v-v^2\delta c] \tag{5.17}$$

$$p_{44} \leftarrow p_{44} + v^2\cdot\sin^2 c\cdot\delta c + \cos^2 c\cdot\delta v \tag{5.18}$$

5.2.2 Actions Dependent on the Filtering Approach

The design parameters in the maneuvre detection algorithm, and the actions following a detection, may very well depend on the tracking approach, i.e., the mathematical models of the system dynamics and the observation, and the version of the filter equations. A "tuning" of the maneuvre detection algorithm to each filtering approach may therefore be necessary, and special actions following a maneuvre detection may be necessary to get the best possible performance for each filtering approach.

In the following we will discuss some of the actions that may be necessary (and feasible) to take for the global iterated filters, and for the parallel filter approach. Only simulation results can tell, however, whether further "tuning" of parameters or special actions can give better performance of an individual filtering approach, so this topic will not at all be exhausted by the following discussion.

5.2.2.1 Global Iterated Filters

The selection of the process noise covariance matrix V_k is among the parameters that influence the "covariance level" of the filter, and thereby the size of the elements of the gain matrix K_k. In fact, the degree of limited memory can, to a certain extent, be controlled by V_k.

Generally, V_k is decided as a compromise between tracking performance when the target is going on straight course and speed, and the filters ability to track small maneuvres (without alerting the maneuvre detection and handling-system).

Now, for the global iterated filters described in sections 3.6.1 and 3.6.2, a better tracking performance can be obtained during nonmaneuvering periods of the target with larger values of the elements of the process noise covariance matrix V_k. The reason for this is the smoothing effect that is a resut of the iterations, tending to stabilize the target velocity vector when the target is moving with constant course and speed.

At the same time, larger values on the elements of V_k will allow the filter to follow small target maneuvres better.

As a result of this, however, the maneuvre detection problem may turn out to be more difficult, since higher values of the elements of the gain-matrix K_k will tend to decrease the value of the innovation signal during target maneuvres, at the same time as the expected variance of the innovation signal will increase due to higher values of the elements of the covariance matrix (see equation (5.1)). An obvious result will be that different values of the design-parameters L, γ and c_k have to be found.

If the iteration interval [(k-N)T, kT], prior to the maneuvre detection, is larger than the interval [kT-Δt, kT], where Δt is the nominal time delay between the occurrence of a maneuvre and its detection, i.e., if NT > Δt, an obvious action to take is to decrease the iteration interval temporarily in such a way that it is contained in [kT-Δt, kT]. (Premaneuvre observations should not be taken into account when post-maneuvre course and speed calculations are performed). Since the observations taken in the time interval [(kT-Δt),kT] comes from a maneuvering target, it might even be interesting to look at the possibility of deviding the iteration interval [(k-N)T, kT], N = Δt/T, into subintervals, so that the observations contained in each sub-interval are more consistent with the constant course and speed hypothesis on which the mathematical model of the system dynamics are built. A realization of this idea in the global iteration context could be a iteration interval [(k_1-N_1)T, k_1T], where N_1 < N, starting with k_1-N_1 = k-N, and finishing when k_1 = k (a sliding iteration "window" over the greater iteration interval [(k-N)T, kT]).

The "serial" filter approach, which is a special case of the global iterated filter (see section 3.6.3), will also need some special treatment.

First, maneuvre detection should be made on the basis of the innovation signal at time k, ε_k. (As can be seen from equation (3.96) this approach has, formally, two observations, z_k and z_{k-N}, and consequently two innovation signals, ε_k and ε_{k-N}).

When a target maneuvre is detected, we propose to reinitialize the filter. It is assumed that we have NT > Δt (this is one of the design criterias for N). The reinitalizing sequence will consist of the following steps:

1. Initialize a single Cartesian Coordinate System filter with

$$\underline{x}_0 = \underline{x}_{2k} = \underline{x}_{k-N} \tag{5.19}$$

and

$$P_0 = P_{22k,k} \tag{5.20}$$

where $P_{22k,k}$ is the lower diagonal part of the covariance matrix for the serial filter, given by:

$$\tilde{P}_{k,k} = \begin{bmatrix} P_{11k,k} & P_{12k,k} \\ P_{21k,k} & P_{22k,k} \end{bmatrix} \tag{5.21}$$

Each of the submatrices in equation (5.21) are 4 by 4 matrices.

2. Run this single filter from time (k-N)T up to time (kT-Δt). (Time kT is the time when the maneuvre was detected).

3. At time $(kT-\Delta t)$, impose limited memory on this single filter as described in section 5.2.1. For reasons of convenience of notation, we now define

$$k_2T = kT-\Delta t \tag{5.22}$$

Then after increasing the velocity elements of the covariance matrix as described in section 5.2.1, the resulting value of the covariance matrix at sample k_2 is given by P_{k_2}. The state vector is given by $\underline{x}_{k_2}$.

4. Run the single filter from time k_2T up to time kT, where a discrete jump in the position and velocity at time kT will be the result (as compared to $\underline{x}_{1k}$ of the serial filter at time of maneuvre detection).

5. Continue to run the single filter on new observations up to time $(k_2+N)T$. Now the full serial filter can be initialized, with the following initial values:

$$\tilde{P}_0 = \begin{bmatrix} P_{k_2+N} & 0 \\ 0 & P_{k_2} \end{bmatrix} \tag{5.23}$$

and

$$\underline{\tilde{x}}_0 = \begin{bmatrix} \underline{x}_{k_2+N} \\ \underline{x}_{k_2} \end{bmatrix} \tag{5.24}$$

5.2.2.2 Parallel Filters

As was proposed in section 3.7, the number of parallel filters can gradually be reduced from M towards 1 as the filters with unlikely ranges are being pruned off. It is therefore obvious that we have to consider two separate cases for maneuvre detection with this approach. namely:

A. Maneuvre detection when the filter with correct range is still not recognized.

B. Maneuvre detection when only the filter with correct range is updated.

In order to save computertime in Case A, we propose to run the maneuvre detection algorithm only on one filter, namely the filter which, at each time instant kT, has the highest probability function $p(\hat{R}_{k_i} = R_k/z_k)$. (Intuitively, it don't seem likely that one of the parallel filters will be better in identifying a maneuvre, since range is not observable. If σ_R is kept low, however, any of the filters should be able to detect a target course/speed change).

For the parallel filter approach, it seems likely that we can give up our claim to take the detection delay Δt into account, without degradation in tracking performance, so this will be proposed. The reasons why this is possible, are the following:

1. After maneuvre detection, independent of case 1 and 2 above, we intend to reinitialize all the M filters, with different ranges centered around:

   Case A: The range calculated from the state vector given by equation (3.110).

   Case B: The range given by the remaining filter. An eventual accumulated range error due to the detection delay Δt, will then be picked up by one of the re-initialized filters.

2. In case A, the range for all the remaining filters may be wrong anyway, even at the time when the maneuvre started.

3. If the time delay should be taken into account in case A, we would have to store the state vector and the covariance matrix for (in the worst case) all the M filters over the last N samples, where $NT \geq \Delta t$. This would claim for a lot of computer memory capacity.

The re-initialization values for the state vector and the covariance matrices for the M-filters should depend on the filter status (case A or B) at the moment of maneuvre detection:

Case A: The center-range filter's state vector is given by equation (3.110). From this state vector, the center range R_0 can be calculated. Then, selecting the range difference between the filters, ΔR_1, each filter's initial range can be calculated from equation (4.63), and the initial position data subsequently from equations (4.64) and (4.65).

The reinitialization values for the velocity components could either be calculated from equations (4.24) and (4.25), or we could simply give all the M filters the same velocity, given by the state vector at the moment of maneuvre detection (equation (3.110)), and let the Kalman filter algorithm adapt the velocity components to each individual filters range.

Case B: Center range is given by the remaining filter. Since, in this case, range is assumed to be more correct, the range difference between each filter after reinitialization should be much less than in case A. With a selected $\Delta R_2 < \Delta R_1$, each filter's initial range can be calculated from equation (4.63), (substitute ΔR_2 for ΔR_1), and the initial position data subsequently from equations (4.64) and (4.65).

Since, in this case, the range difference between each filter is smaller, the remaining filter's velocity components can be used as initial values for all the M filters.

Concerning the re-initialization values for the M covariance matrices, we propose to consider two possibilities:

1. Use the initialization procedure described in section 4.6.2.
2. Use the covariance matrix for the remaining filter (case B), or for the filter with highest $p(\hat{R}_{ki} = R/z_k)$ (case A), but modify its elements in the following way:

$$p_{11_i} \leftarrow p_{11_0} + \cos^2\phi\cdot\sigma^2_{\phi_0}\cdot(R_i^2-R_0^2)$$

$$p_{12_i} \leftarrow p_{12_0} - \sin\phi_0 \cos\phi_0\cdot\sigma^2_{\phi_0}(R_i^2-R_0^2)$$

$$p_{21_i} \leftarrow p_{21_0} - \sin\phi_0 \cos\phi_0\cdot\sigma^2_{\phi_0}(R_i^2-R_0^2)$$

$$p_{22_i} \leftarrow p_{22_0} + \sin^2\phi_0\cdot\sigma^2_{\phi_0}(R_i^2-R_0^2)$$

where R_0, p_{11_0}, p_{12_0}, p_{21_0}, p_{22_0} and ϕ_0 are referring to the center filter, and R_i, p_{11_i}, p_{12_i}, p_{21_i} and p_{22_i}, $i = 1,2,\ldots,M$ are the initialization values of the position elements of the covariance matrices for the M parallel filters.

In addition to this , impose limited memory on the filters by increasing the velocity elements of the covariance matrices as described in section 5.2.1.

Which of the two proposed methods for covariance matrix re-initialization that should be selected, has to be decided upon through simulations, as the procedure giving the best result.

6. SIMULATION GUIDELINES

The development of filtering approaches, initialization routines and maneuvering target tracking approaches given in the preceding chapters, have to be verified and compared through an extensive simulation study.

Each algorithm should be optimized with respect to design parameters like process noise variance, maneuvre detection parameters and actions, etc., before comparison between the different approaches can be performed.

In order to arrive at correct conclusions, the target-observer geometry used in the simulations are extremely important. On the other hand, in order to reduce the number of simulation runs to an acceptable level, only a few different geometries should be considered.

The simulation study should further focus on two different problems:

1. The initialization phase, each filters ability to achieve correct range estimate as fast as possible.
2. Maneuvre detection, ability to track a target through different maneuvres (small/large course/speed changes).

The simulation results should be visualized on plots giving range error, velocity error and course error as a function of time. x,y-plots should also be provided.

It is believed not to be necessary to run Monte Carlo simulation studies on all the 10 different approaches given in Chapter 3. A few single simulation runs for each filter approach for a small number of target-observer geometry should reveal which approaches are worth further study.

The most promising 2 or 3 approaches should then finally be compared through Monte Carlo simulations.

REFERENCES

[1] D.J. Murphy: Noisy Bearings - Only Target Motion Analysis. Ph.D. Diss., Dept. Electrical Engineering, Northeastern Univ., 1970.

[2] A.H. Jazwinski: Stochastic Processes and Filtering Theory. Academic Press 1970.

[3] F.R. Castella, F.G. Dunnebacke: Analytical Results for the X,Y-Kalman Tracking Filter. IEEE Transactions on Aerospace and Electronic Systems, November 1974.

[4] R.R. Tenney, R.S.Hebbert, N.R. Sandell, Jr.: Tracking a Maneuvering Acoustical Source with a Single Frequency-Bearing Sensor. Proc. IEEE 1976 Conference on Decision and Control.

[5] D.R. Morgan: A Target Trajectory Noise Model for Kalman Trackers. IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-12, No. 3, May 1976.

[6] K.V. Ramachandra, U.S. Srinivasan: Steady State Results for the X,Y,Z Kalman Tracking Filter. IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-13, July 1977.

[7] Allen G. Lindgren, Kai F. Gong: Position and Velocity Estimation Via Bearing Observations. IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-14, No. 4, July 1978.

[8] A. Farina, S. Pardini: Track-While-Scan Algorithm In a Clutter Environment. IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-14, No. 5, September 1978.

[9] Y.T. Chan, A.G.C. Hu, J.B. Plant: A Kalman Filter Based Tracking Scheme with Input Estimation. IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-15, No. 2, March 1979.

[10] James S. Thorp: Optimal Tracking of Maneuvering Targets. IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-9, No. 4, July 1973.

[11] Vincent J. Aidala: Kalman Filter Behaviour in Bearings-Only Tracking Applications. IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-15, No. 1, January 1979.

[12] Arthur Gelb: Appplied Optimal Estimation. MIT Press 1974.

[13] R.P. Wishner, J.A. Tabaczynski, and M. Athans: A Comparison of Three Non-Linear Filters. Automatica, Vol. 5, pp. 487-496, 1969.

[14] George G. Ricker, Jack R. Williams: Adaptive Tracking Filter for Maneuvering Targets. IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-14, No. 1, Jan. 1978.

[15] Raman K. Mehra: On the Identification of Variances and Adaptive Kalman Filtering. IEEE Trans. on Automatic Control, Vol. AC-15, No. 2, April 1970.

[16] Raman K. Mehra: Approaches to Adaptive Filtering. IEEE Trans. on Automatic Control, Oct. 1972.

[17] Leonard Chin: The Impact of Adaptive Kalman Filter on Inertial Navigation Systems Design. Agard Conference Sandefjord, May 1978.

[18] Leonard Chin: Advanced in Adaptive Filtering. In: Control and Dynamic Systems, Vol. 15, 1979. Edited by: C.T. Leondes.

[19] N.H. Gholson, E.L. Moose: Maneuvering Target Tracking Using Adaptive State Estimation. IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-13, No. 3, May 1972.

[20] V.J. Aidala, J.S. Davis: The Utilization of Data Measurement Residuals for Adaptive Kalman Filtering. Proc. of the IEEE Conference Engineering in the Ocean Environment 1973 (Ocean 1973).

[21] C.M. Brown, Jr., C.F. Price: A Comparison of Adaptive Tracking Filters for Targets of Variable Maneuverability. Proc. IEEE 1976 Conference on Decision and Control.

[22] R.L. Moose, H.F. Vanlandingham, D.H. McCabe: Modeling and Estimation for Tracking Maneuvering Targets. IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-15, No. 3, May 1979.

[23] Peter S. Maybeck: Stochastic Models, Estimation and Control. Volume 1. Academic Press 1979.

[24] R.R. Tenney, T.B. Ballard, L.E. Miller: A Dual Passive Tracking Filter for Maneuvering Targets. Proc. of the Conference on Advances in Passive Target Tracking, Monterey, California, May 1977.

[25] A Jazwinski: Limited Memory Optimal Filtering. IEEE Trans. on Automatic Control, October 1968.

[26] A.S. Willsky, H.L. Jones: A Generalized Likelihood Ratio Approach to State Estimation in Linear Systems Subject to Abrupt Changes. Proc. 1974 IEEE Conference on Decision and Control, November 1974.

[27] A.S. Willsky, H.L. Jones: A Generalized Likelihood Ratio Approach to the Detection and Estimation of Jumps in Linear Systems. IEEE Trans. on Automatic Control, February 1976.

[28] R.L. Moose: An Adaptive State Estimation Solution to the Maneuvering Target Problem. IEEE Trans. on Automatic Control, June 1975.

[29] R.L. Moose, P.P. Wang: An Adaptive Estimator with Learning for a Plant Containing Semi-Markov Switching Parameters. IEEE Trans. on Systems, Man, and Cybernetics, May 1973.

[30] H.L. Van Trees: Detection, Estimation and Modulation Theory, Part 1. Wiley and Sons, 1968.

APPENDIX A

LINEARIZATION OF $\underline{f}(\underline{x}_k)$

We have

$$\underline{f}(\underline{x}_k^p) = \begin{bmatrix} \phi_k + \tan^{-1}\left(\dfrac{\Delta L_{Nk}}{R_k + \Delta L_{pk}}\right) \\ \sqrt{(R_k+\Delta L_{pk})^2 + \Delta L_{Nk}^2} \\ v_{xk} \\ v_{yk} \end{bmatrix} \tag{A1}$$

We intend to calculate

$$F(\underline{x}_k^p) = \frac{\partial f(\underline{x}_k^p)}{\partial \underline{x}_k^p} = \begin{bmatrix} \dfrac{\partial f_1}{\partial \phi} & \dfrac{\partial f_1}{\partial R} & \dfrac{\partial f_1}{\partial v_x} & \dfrac{\partial f_1}{\partial v_y} \\ \dfrac{\partial f_2}{\partial \phi} & \dfrac{\partial f_2}{\partial R} & \dfrac{\partial f_2}{\partial v_x} & \dfrac{\partial f_2}{\partial v_y} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{A2}$$

In order to be able to calculate the different elements of $F(\underline{x}_k^p)$, the following derivatives are needed:

$$\frac{\partial R_k}{\partial \phi_k} = 0 \tag{A3}$$

$$\frac{\partial \Delta L_{Nk}}{\partial \phi_k} = -\Delta L_{pk} \tag{A4}$$

$$\frac{\partial \Delta L_{pk}}{\partial \phi_k} = \Delta L_{Nk} \tag{A5}$$

$$\frac{\partial R_k}{\partial v_{xk}} = 0 \tag{A6}$$

$$\frac{\partial R_k}{\partial v_{yk}} = 0 \tag{A7}$$

$$\frac{\partial \Delta L_{Nk}}{\partial v_{xk}} = T \cdot \cos\phi_k \tag{A8}$$

$$\frac{\partial \Delta L_{pk}}{\partial v_{xk}} = T \sin\phi_k \tag{A9}$$

$$\frac{\partial \Delta L_{Nk}}{\partial v_{yk}} = -T \sin\phi_k \tag{A10}$$

$$\frac{\partial \Delta L_{pk}}{\partial v_{yk}} = T \cos\phi_k \tag{A11}$$

Next, we define the following two variables:

$$\Delta L_{xk} = (R_k + \Delta L_{pk})\sin \phi_k + \Delta L_{Nk} \cdot \cos \phi_k \tag{A12}$$

$$\Delta L_{yk} = (R_k + \Delta L_{pk})\cos \phi_k - \Delta L_{Nk} \sin \phi_k \tag{A13}$$

Then we have:

$$\frac{\partial f_1}{\partial \phi_k} = 1 - \frac{\Delta L_{pk}(R_k+\Delta L_{pk}) + \Delta L_{Nk}^2}{R_{k+1}^2} = \frac{R_k(R_k+\Delta L_{pk})}{R_{k+1}^2} \tag{A14}$$

$$\frac{\partial f_1}{\partial R_k} = -\frac{\Delta L_{Nk}}{R_{k+1}^2} \tag{A15}$$

$$\frac{\partial f_1}{\partial v_{xk}} = \frac{T[(R_k+\Delta L_{pk})\cos\phi_k - \Delta L_{Nk}\sin\phi_k]}{R_{k+1}^2} = \frac{T\Delta L_{yk}}{R_{k+1}^2} \tag{A16}$$

$$\frac{\partial f_1}{\partial v_{yk}} = \frac{-T[(R_k+\Delta L_{pk})\sin\phi_k + \Delta L_{Nk}\cos\phi_k]}{R_{k+1}^2} = \frac{-T\Delta L_{xk}}{R_{k+1}^2} \tag{A17}$$

$$\frac{\partial f_2}{\partial \phi_k} = \frac{R_k \cdot \Delta L_{Nk}}{R_{k+1}} \tag{A18}$$

$$\frac{\partial f_2}{\partial R_k} = \frac{R_k+\Delta L_{pk}}{R_{k+1}} \tag{A19}$$

$$\frac{\partial f_2}{\partial v_{xk}} = \frac{T[(R_k+\Delta L_{pk})\sin\phi_k + \Delta L_{Nk}\cos\phi_k]}{R_{k+1}} = \frac{T\cdot\Delta L_{xk}}{R_{k+1}} \tag{A20}$$

$$\frac{\partial f_2}{\partial v_{yk}} = \frac{T[(R_k+\Delta L_{pk})\cos\phi_k + \Delta L_{Nk}\sin\phi_k]}{R_{k+1}} = \frac{T\Delta L_{yk}}{R_{k+1}} \tag{A21}$$

Equation (A14)-(A21) can now be inserted in eq. (A2), and we have our final result:

$$F(\underline{x}_k^p) = \begin{bmatrix} \frac{R_k(R_k+\Delta L_{pk})}{R_{k+1}^2} & \frac{-\Delta L_{Nk}}{R_{k+1}^2} & \frac{T\Delta L_{yk}}{R_{k+1}^2} & \frac{-T\Delta L_{xk}}{R_{k+1}^2} \\ \frac{R_k\cdot\Delta L_{Nk}}{R_{k+1}} & \frac{R_k+\Delta L_{pk}}{R_{k+1}} & \frac{T\Delta L_{xk}}{R_{k+1}} & \frac{T\Delta L_{yk}}{R_{k+1}} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{A22}$$

APPENDIX B

CALCULATION OF THE ADDITIONAL TERMS $\underset{\rightarrow}{c}_k$ AND L_k FOR THE SECOND ORDER GAUSSIAN POLAR COORDINATE SYSTEM MODEL FILTER.

1. CALCULATION OF $\underset{\rightarrow}{c}_k$

$$c_{ki} = \text{trace}\left\{\frac{\partial}{\partial \underset{\rightarrow}{x}_k^p}\left[\frac{\partial f_i}{\partial \underset{\rightarrow}{x}_k^p}\right]^T \cdot P_{k,k}\right\}, \; i=1,\ldots 4 \tag{B1}$$

Now, $[\partial f_i/\partial \underset{\rightarrow}{x}_k^p]$ has already been calculated in APPENDIX A. We have:

$$F(\underset{\rightarrow}{x}_k^p) = \begin{bmatrix} \dfrac{\partial f_1}{\partial \underset{\rightarrow}{x}_k^p} \\[2ex] \dfrac{\partial f_2}{\partial \underset{\rightarrow}{x}_k^p} \\[2ex] \dfrac{\partial f_3}{\partial \underset{\rightarrow}{x}_k^p} \\[2ex] \dfrac{\partial f_4}{\partial \underset{\rightarrow}{x}_k^p} \end{bmatrix} \begin{bmatrix} \dfrac{R_k(R_k+\Delta L_{pk})}{R_{k+1}^2} & \dfrac{-\Delta L_{Nk}}{R_{k+1}^2} & \dfrac{T\Delta L_{yk}}{R_{k+1}^2} & \dfrac{-T\Delta L_{xk}}{R_{k+1}^2} \\[2ex] \dfrac{R_k \Delta L_{Nk}}{R_{k+1}} & \dfrac{R_k+\Delta L_{pk}}{R_{k+1}} & \dfrac{T\Delta L_{xk}}{R_{k+1}} & \dfrac{T\Delta L_{yk}}{R_{k+1}} \\[2ex] 0 & 0 & 1 & 0 \\[2ex] 0 & 0 & 0 & 1 \end{bmatrix} \tag{B2}$$

Next, we define:

$$F1 = \frac{\partial}{\partial \underset{\rightarrow}{x}_k^p}\left[\frac{\partial f_1}{\partial \underset{\rightarrow}{x}_k^p}\right]^T = \frac{\partial}{\partial \underset{\rightarrow}{x}_k^p} \begin{bmatrix} R_k(R_k+\Delta L_{pk})/R_{k+1}^2 \\ -\Delta L_{Nk}/R_{k+1}^2 \\ T\Delta L_{yk}/R_{k+1}^2 \\ -T\Delta L_{xk}/R_{k+1}^2 \end{bmatrix} \tag{B3}$$

$$F2 = \frac{\partial}{\partial \underline{x}_k^p} \left[\frac{\partial f_2}{\partial \underline{x}_k^p}\right]^T = \frac{\partial}{\partial \underline{x}_k^p} \begin{bmatrix} R_k \Delta L_{Nk}/R_{k+1} \\ (R_k + \Delta L_{pk})/R_{k+1} \\ T\Delta L_{xk}/R_{k+1} \\ T\Delta L_{yk}/R_{k+1} \end{bmatrix} \tag{B4}$$

The elements of the two 4.4-matrices F1 and F2 are calculated subsequently:

$$f1_{11} = -\frac{\Delta L_{Nk} \cdot R_k}{R_{k+1}^4} [R_k^2 - \Delta L_{pk}^2 - \Delta L_{Nk}^2] \tag{B5}$$

$$f1_{12} = \frac{1}{R_{k+1}^4} \cdot [\Delta L_{pk} \cdot R_{k+1}^2 + 2\Delta L_{Nk}^2 \cdot R_k] \tag{B6}$$

$$f1_{13} = -\frac{T \cdot R_k}{R_{k+1}^4} [(R_{k+1}^2 - 2\Delta L_{Nk}^2) \sin \phi_k + 2\Delta L_{Nk}(R_k + \Delta L_{pk}) \cos \phi_k] \tag{B7}$$

$$f1_{14} = \frac{T \cdot R_k}{R_{k+1}^4} [-(R_{k+1}^2 - 2\Delta L_{Nk}^2) \cos \phi_k + 2\Delta L_{Nk}(R_k + L_{pk}) \sin \phi_k] \tag{B8}$$

$$f1_{21} = \frac{1}{R_{k+1}^4} [\Delta L_{pk} R_{k+1}^2 + 2\Delta L_{Nk}^2 \cdot R_k] \tag{B9}$$

$$f1_{22} = \frac{2\Delta L_{Nk}}{R_{k+1}^4} [R_k + \Delta L_{pk}] \tag{B10}$$

$$f1_{23} = -\frac{T}{R_{k+1}^4} [R_{k+1}^2 \cdot \cos \phi_k - 2\Delta L_{Nk} \Delta L_{xk}] \tag{B11}$$

$$f1_{24} = \frac{T}{R_{k+1}^4}[R_{k+1}^2 \cdot \sin\phi_k + 2\Delta L_{Nk}\Delta L_{yk}] \tag{B12}$$

$$f1_{31} = -R_k \cdot f1_{24} \tag{B13}$$

$$f1_{32} = \frac{T}{R_{k+1}^4}[R_{k+1}^2 \cdot \cos\phi_k + 2\Delta L_{yk}(R_k + \Delta L_{pk})] \tag{B14}$$

$$f1_{33} = -\frac{2T^2\Delta L_{yk}\Delta L_{xk}}{R_{k+1}^4} \tag{B15}$$

$$f1_{34} = \frac{T^2}{R_{k+1}^4}[R_{k+1}^2 + 2\Delta L_{yk}^2] \tag{B16}$$

$$f1_{41} = R_k \cdot f1_{23} \tag{B17}$$

$$f1_{42} = -\frac{T}{R_{k+1}^4}[R_{k+1}^2 \sin\phi_k - 2\Delta L_{xk}(R_k + \Delta L_{pk})] \tag{B18}$$

$$f1_{43} = -\frac{T^2}{R_{k+1}^4}[R_{k+1}^2 - 2\Delta L_{xk}\Delta L_{yk}] \tag{B19}$$

$$f1_{44} = -f1_{33} \tag{B20}$$

$$f2_{11} = -\frac{R_k}{R_{k+1}^3}\left[\Delta L_{pk} R_{k+1}^2 + \Delta L_{Nk}^2 \cdot R_k\right] \tag{B21}$$

$$f2_{12} = \frac{\Delta L_{Nk}}{R_{k+1}^3}\left|R_{k+1}^2 - \Delta L_{Nk} \cdot R_k^2\right| \tag{B22}$$

$$f2_{13} = -\frac{T \cdot R_k}{R^3_{k+1}} \, [R^2_{k+1} \cos \phi_k - \Delta L_{Nk} \Delta L_{xk}] \tag{B23}$$

$$f2_{14} = -\frac{TR_k}{R^3_{k+1}} \, [R^2_{k+1} \sin \phi_k + \Delta L_{Nk} \Delta L_{yk}] \tag{B24}$$

$$f2_{21} = \frac{1}{R^3_{k+1}} \, [\Delta L_{pk} R^2_{k+1} - R_k \Delta L_{Nk} (R_k + \Delta L_{pk})] \tag{B25}$$

$$f2_{22} = \frac{\Delta L^2_{Nk}}{R^3_{k+1}} \tag{B26}$$

$$f2 = \frac{T}{R^3_{k+1}} \, [R^2_{k+1} \sin \phi_k - (R_k + \Delta L_{pk}) \Delta L_{xk}] \tag{B27}$$

$$f2_{24} = \frac{T}{R^3_{k+1}} \, [R^2_{k+1} \sin \phi_k - (R_k + \Delta L_{pk}) \Delta L_{yk}] \tag{B28}$$

$$f2_{31} = f2_{13} \tag{B29}$$

$$f2_{32} = f2_{23} \tag{B30}$$

$$f2_{33} = \frac{T^2}{R^3_{k+1}} \, [R^2_{k+1} - \Delta L^2_{xk}] \tag{B31}$$

$$f2_{34} = -\frac{T^2 \Delta L_{xk} \cdot \Delta L_{yk}}{R^3_{k+1}} \tag{B32}$$

$$f2_{41} = f2_{14} \tag{B33}$$

$$f2_{42} = f2_{24} \tag{B34}$$

$$f3_{43} = f2_{34} \tag{B35}$$

$$f2_{44} = \frac{T^2 [R^2_{k+1} - \Delta L^2_{yk}]}{R^3_{k+1}} \tag{B36}$$

Now, having calculated all the elements in the two matrices F1 and F2, the vector $\underset{\sim}{c}_k$ can be calculated from equation (B1). The result:

$$\underset{\sim}{c}_k = \begin{bmatrix} \sum\limits_{i,j=1}^{4} f1_{ji} \cdot P_{ij} \\ \sum\limits_{i,j=1}^{4} f2_{ji} P_{ij} \\ 0 \\ 0 \end{bmatrix} \tag{B37}$$

where P_{ij} is the $(i,j)^{th}$ element of the covariance matrix $P_{k,k}$.

2. CALCULATION OF L_k

The matrix L_k is defined by:

$$L_k = \frac{1}{2} (\partial^2 f P^2 \partial^2 f) \tag{B38}$$

In order to be able to calculate L_k, we first define te two matrices S and T:

$$S = \begin{bmatrix} \sum_{i=1}^{4} fl_{1i} P_{i1} & \sum_{i=1}^{4} fl_{1i} P_{i2} & \cdots & \sum_{i=1}^{4} fl_{1i} P_{i4} \\ \vdots & & \ddots & \vdots \\ \sum_{i=1}^{4} fl_{4i} P_{i1} & \cdots & \cdots & \sum_{i=1}^{4} fl_{4i} P_{i4} \end{bmatrix} \tag{B39}$$

$$T = \begin{bmatrix} \sum_{i=1}^{4} f2_{1i} P_{i1} & \cdots & \sum_{i=1}^{4} f2_{1i} P_{i4} \\ \vdots & \ddots & \\ \sum_{i=1}^{4} f2_{4i} P_{i1} & \cdots & \sum_{i=1}^{4} f2_{4i} P_{i4} \end{bmatrix} \tag{B40}$$

Then the 4 nonzero elements of the matrix L_k are given by:

$$\ell_{11} = \frac{1}{2} \text{trace } (S \cdot S) \tag{B41}$$

$$\ell_{12} = \frac{1}{2} \text{trace } (S.T) \tag{B42}$$

$$\ell_{21} = \frac{1}{2} \text{trace } (T.S) \tag{B43}$$

$$\ell_{22} = \frac{1}{2} \text{trace } (T.T) \tag{B44}$$

Finally, the matrix L_k is then given by:

$$L_k = \frac{1}{2}\begin{bmatrix} \sum_{j,i=1}^{4} s_{ji}s_{ij} & \sum_{j,i=1}^{4} s_{ji}t_{ij} & 0 & 0 \\ \sum_{j,i=1}^{4} t_{ji}s_{ij} & \sum_{j,i=1}^{4} t_{ji}t_{ij} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \qquad \text{(B45)}$$

APPENDIX C

ITERATED EXTENDED KALMAN FILTER. DETAILED EQUATIONS.

The Extended Kalman filter-equations for the Cartesian Coordinate system representation of our tracking problem are given by equations (3.11)-(3.18). Equations (3.13) and (3.14) are obtained from the following equation for the linearized system:

$$\delta\hat{\underline{x}}_{k,k} = \delta\hat{\underline{x}}_{k,k-1} + K_k(\delta z_k - H_k \delta\hat{\underline{x}}_{k,k-1}) \tag{C1}$$

where

$$\delta\hat{\underline{x}}_{k,k} = \hat{\underline{x}}_{k,k} - \bar{\underline{x}}_k \tag{C2}$$

$$\delta\hat{\underline{x}}_{k,k-1} = \phi(T)\cdot\delta\hat{\underline{x}}_{k-1,k-1} \tag{C3}$$

$$\delta z_k = z_k - \hat{z}_{k,k-1} \tag{C4}$$

The Extended Kalman-filter equation (3.13) is developed from equation (C1) through the special selection of linearization trajectory, namely:

$$\bar{\underline{x}}_k = \hat{\underline{x}}_{k,k-1} \tag{C6}$$

When the observation z_k is processed, $\hat{\underline{x}}_{k,k}$ is obtained, and the system is relinearized about $\hat{\underline{x}}_{k,k}$. Then, after processing of z_k and relinearization, $\delta\hat{\underline{x}}_{k,k} = 0$, and also $\delta\hat{\underline{x}}_{k,k-1} = 0$ in view of equation (C3). As we can see, equation (C1) reduces to equations (3.13) and (3.14).

However, the Extended Kalmanfilter does not utilize the improved linearization trajectory $\hat{\underline{x}}_{k,k}$ for the processing of the observation z_k. This is done by the iteration scheme described by Fig. C1. The $\underline{\varepsilon}$-vector has to be decided upon through simulations.

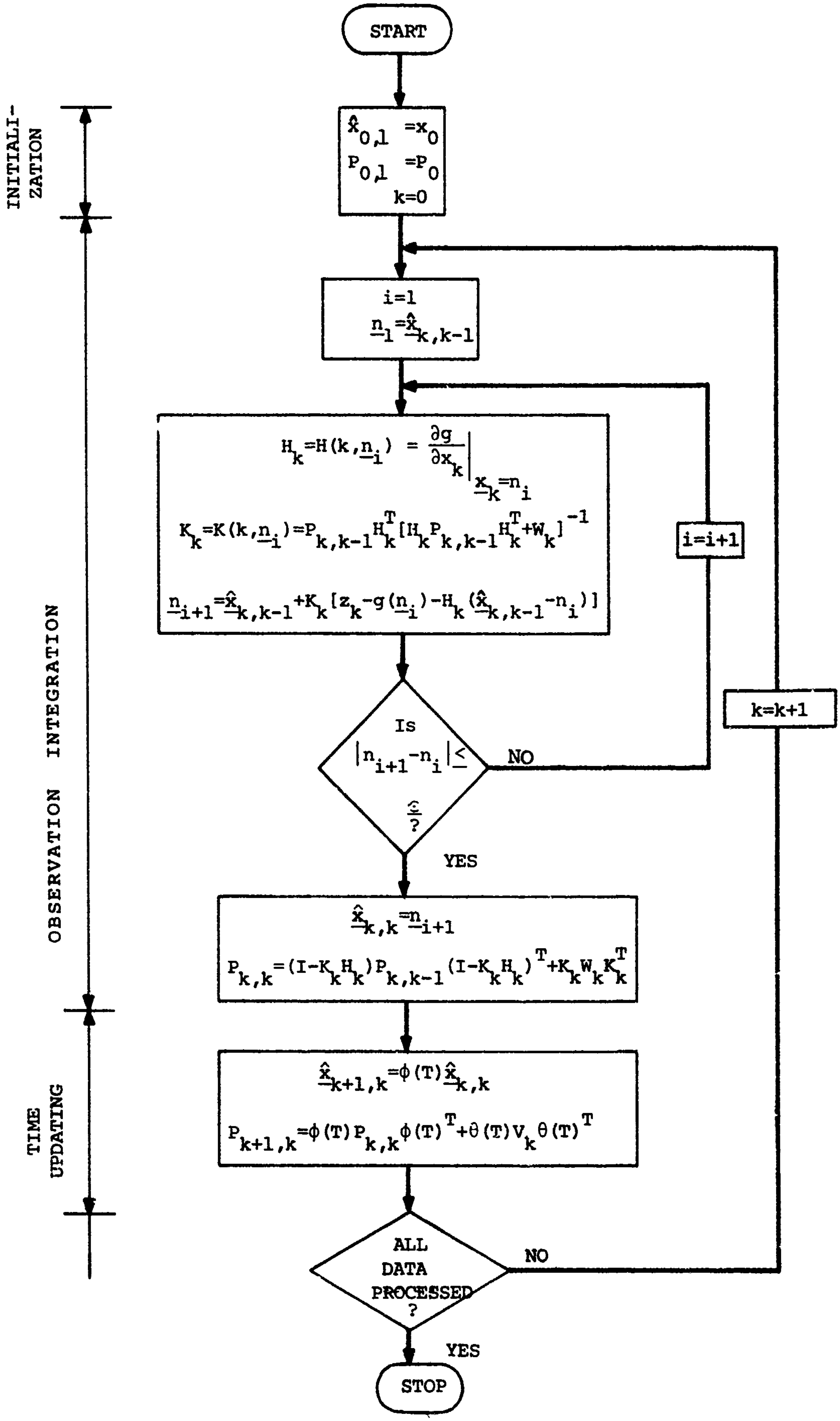


FIG. C1. ITERATED EXTENDED KALMANFILTER

APPENDIX D

ITERATED LINEAR FILTER-SMOOTHER. DETAILED EQUATIONS.

This iteration scheme is adapted to the Polar Coordinate system representation of our tracking problem, where the observation equation is linear. That means some simplifications imposed on the iterator proposed by Jazwinski [2].

In order to avoid the matrix invertion necessary in Jazwinski's approach, the equation

$$\underline{\varepsilon}_{i+1} = \underline{\hat{x}}_{k,k} + S(k, \underline{\varepsilon}_i)[\underline{n}_{i+1} - \underline{\hat{x}}_{k+1,k}] \tag{D1}$$

can be transformed, making it unnecessary to calculate $S(k, \varepsilon_i)$ explicitly. We have:

$$s(k, \varepsilon_i) = P_{k,k}F(k, \varepsilon_i)P^{-1}_{k+1,k} \tag{D2}$$

$$\underline{n}_{i+1} = \underline{\hat{x}}_{k+1,k} + K(k+1, \varepsilon_i)[z_{k+1} - \hat{z}_{k+1,k}] \tag{D3}$$

where

$$K(k+1, \underline{\varepsilon}_i) = P_{k+1,k}H^T(HP_{k+1,k}H^T + W_{k+1})^{-1} \tag{D4}$$

Now, if we define:

$$L(k+1, \underline{\varepsilon}_i) = H^T(HP_{k+1,k}H^T + w_{k+1})^{-1} \tag{D5}$$

which gives:

$$K(k+1, \varepsilon_i) = P_{k+1,k}L(k+1, \varepsilon_i) \tag{D6}$$

equation (D1) can be written, inserting for $\underline{n}_{i+1}$ and $s(k, \varepsilon_i)$:

$$\underline{\varepsilon}_{i+1} = \underline{\hat{x}}_{k,k} + P_{k,k}F(k,\varepsilon_i)L(k+1,\varepsilon_i)[z_{k+1}-\hat{z}_{k+1,k}] \qquad \text{(D7)}$$

The resulting iterator is summarized on Fig. D1.

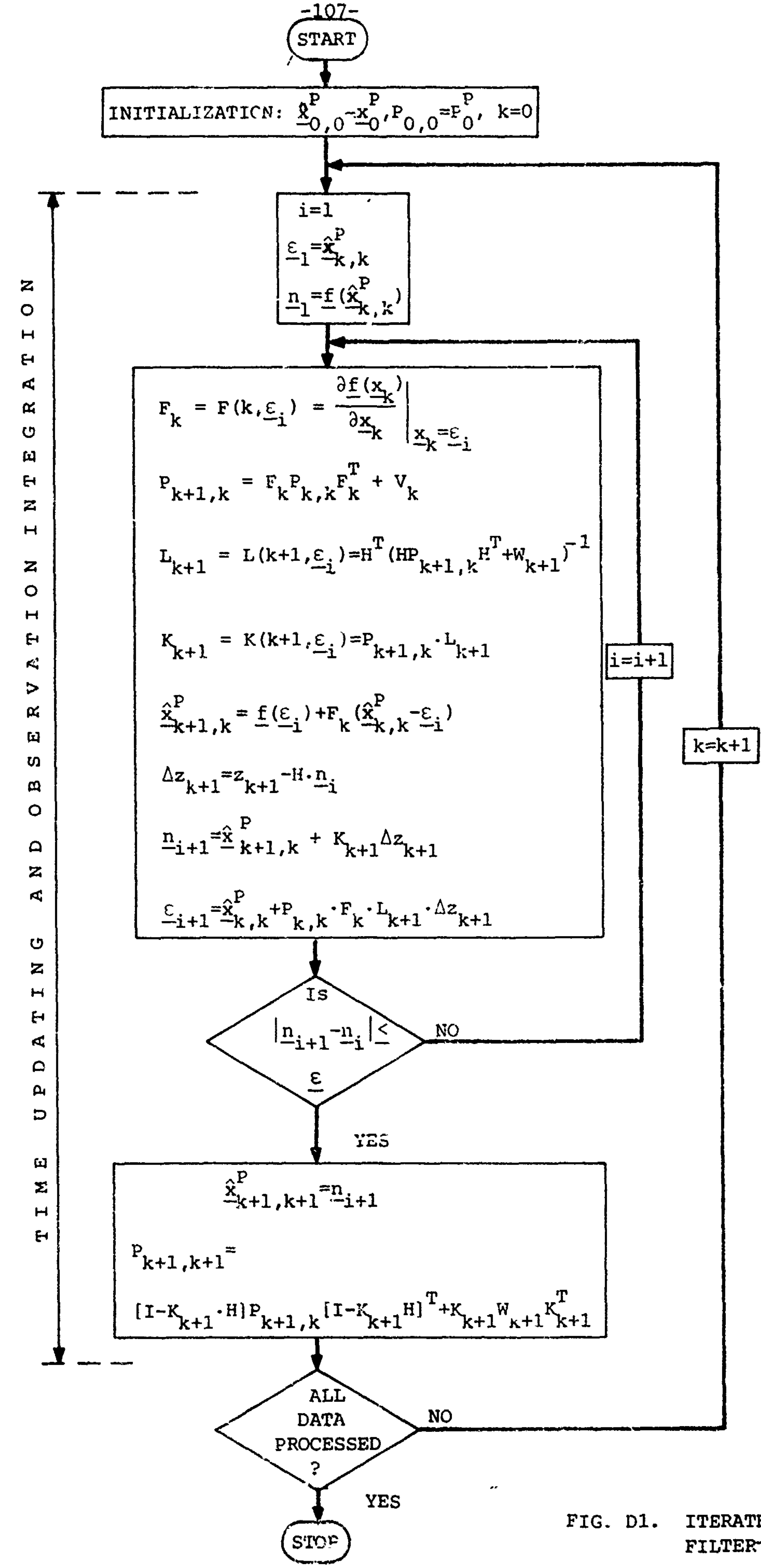


FIG. D1. ITERATED LINEAR FILTER-SMOOTHER

APPENDIX E

DETAILED DERIVATION OF THE COVARIANCE EQUATIONS FOR THE CARTESIAN KALMAN FILTER.

In this Appendix we intend to derive the scalar equations for the a priori and the a posteriori covariance imbedded in the matrix equations (3.18) and (3.16).

The purpose is, if possible, to assimilate a deeper understanding of the Kalman filter mechanism generating the expected variance on the innovation signal.

In the following, we intend to utilize our knowledge of symmetry in the covariance matrix to reduce the number of scalar equations from 16 to 10 equations.

Under these circumstances equation (3.16) reduces from the Joseph form to the form:

$$P_{k,k} = (I-K_kH_k)P_{k,k-1} \tag{E1}$$

where K_k and H_k is given by:

$$K_k = P_{k,k-1}H_k^T(H_kP_{k,k-1}H_k^T+W_k)^{-1} \tag{E2}$$

$$H_k = \begin{bmatrix} \dfrac{\cos\phi_k}{R_k} & -\dfrac{\sin\phi_k}{R_k} & 0 & 0 \end{bmatrix} \tag{E3}$$

The variance of the innovation signal is given by:

$$\sigma_k^2 = H_kP_{k,k-1}H_k^T + W_k \tag{E4}$$

By inserting for H_k from equation (E3) and utilize the symmetry fact of the covariance matrix, equation (E4) can be transformed to:

$$\sigma_k^2 = \frac{1}{R}[P_{11} \cos^2\phi + P_{22}\sin^2\phi - P_{12} \sin 2\phi] + w_k \tag{E5}$$

where the subscript k,k-1 are dropped on the elements of the covariance matrix, and on R and ϕ. The same is done in the following.

Now, equation (E2) can be written as:

$$K_k = P_{k,k-1}H_k^T \cdot \frac{1}{\sigma_k^2} \tag{E6}$$

Next we define:

$$N_k = R^2 \cdot \sigma_k^2 \tag{E7}$$

Then the elements of the K_k-matrix can be calculated to be:

$$K_{1k} = \frac{R}{N_k} [P_{11} \cos \phi - P_{12} \sin\phi] \tag{E8}$$

$$K_{2k} = \frac{R}{N_k} [P_{12} \cos \phi - P_{22} \sin \phi] \tag{E9}$$

$$K_{3k} = \frac{R}{N_k} [P_{13} \cos \phi - P_{23} \sin \phi] \tag{E10}$$

$$K_{4k} = \frac{R}{N_k}[P_{14} \cos \phi - P_{24} \sin \phi] \tag{E11}$$

Now, from equation (E6) we have:

$$H_k P_{k,k-1} = \sigma_k^2 \cdot K_k^T \tag{E12}$$

since the covariance matrix is symmetric.

Then, inserting from equation (E12) in equation (E1) we get:

$$P_{k,k} = P_{k,k-1} - \sigma_k^2 \cdot K_k \cdot K_k^T \tag{E13}$$

Based on equation (E13) and equations (E7)-(E11) we get the following 10 scalar covariance equations:

$$P_{11}^{k,k} = P_{11}^{k,k-1} - \frac{1}{N_k} [P_{11} \cos \phi - P_{12} \sin \phi]^2 \tag{E14}$$

$$P_{12}^{k,k} = P_{12}^{k,k-1} - \frac{1}{N_k} [P_{11} \cos \phi - P_{12} \sin \phi] [P_{12} \cos \phi - P_{22} \sin \phi] \tag{E15}$$

$$P_{13}^{k,k} = P_{13}^{k,k-1} - \frac{1}{N_3} [P_{11} \cos \phi - P_{12} \sin \phi] [P_{13} \cos \phi - P_{23} \sin \phi] \tag{E16}$$

$$P_{14}^{k,k} = P_{14}^{k,k-1} - \frac{1}{N_k} [P_{11} \cos \phi - P_{12} \sin \phi] [P_{14} \cos \phi - P_{24} \sin \phi] \tag{E17}$$

$$P_{22}^{k,k} = P_{22}^{k,k-1} - \frac{1}{N_k} [P_{12} \cos \phi - P_{22} \sin \phi]^2 \tag{E18}$$

$$P_{23}^{k,k} = P_{23}^{k,k-1} - \frac{1}{N_k} [P_{12} \cos \phi - P_{22} \sin \phi] [P_{13} \cos \phi - P_{23} \sin \phi] \tag{E19}$$

$$P_{24}^{k,k} = P_{24}^{k,k-1} - \frac{1}{N_k} [P_{12} \cos \phi - P_{22} \sin \phi] [P_{14} \cos \phi - P_{24} \sin \phi] \tag{E20}$$

$$P_{33}^{k,k} = P_{33}^{k,k-1} - \frac{1}{N_k} [P_{13} \cos \phi - P_{23} \sin \phi]^2 \tag{E21}$$

$$P_{34}^{k,k} = P_{34}^{k,k-1} - \frac{1}{N_k} [P_{13} \cos \phi - P_{23} \sin \phi] [P_{14} \cos \phi - P_{24} \sin \phi] \tag{E22}$$

$$P_{44}^{k,k} = P_{44}^{k,k-1} - \frac{1}{N_4} [P_{14} \cos \phi - P_{24} \sin \phi]^2 \tag{E23}$$

In equations (E14)-(E23), most of the variables on the right hand side of the equations have got their subscript k,k-1 dropped.

Now, the only place the range R enters the equations (E14)-(E23), is through N_k. From equation (E7) and (E5) N_k can be given by:

$$N_k = P_{11} \cos^2\phi + P_{22} \sin^2\phi - P_{12} \sin 2\phi + R^2 W_k \tag{E24}$$

Equivalently, if we define:

$$\sigma_R^2 = P_{11} \sin^2\phi + P_{22} \cos^2\phi + P_{12} \sin 2\phi \tag{E25}$$

equation (E24) can be written:

$$N_k = P_{11} + P_{22} - \sigma_R^2 + R^2 W_k \tag{E26}$$

Equations (E14) through (E23) gives the a posteriori covariance after an observation intergration, based on the a priori covariance prior to the observation integration.

Timeupdating of the covariance, i.e., calculating the a priori covariance at the next sample, based on the a posteriori variance at the current sample, is performed through equation (3.18). By multiplying out this matrix equation, utilizing the symmetric properties of the covariance matrix, we get the following 10 time updating equations:

$$P_{11}^{k+1,k} = P_{11}^{k,k} + 2T\, P_{13}^{k,k} + T^2\, P_{13}^{k,k} + (T^4/4)\cdot V_{1k} \tag{E27}$$

$$P_{12}^{k+1,k} = P_{12}^{k,k} + T(P_{23}^{k,k} + P_{14}^{k,k}) + T^2 P_{34}^{k,k} \tag{E28}$$

$$P_{13}^{k+1,k} = P_{13}^{k,k} + T\, P_{33}^{k,k} + (T^3/2)\cdot V_{1k} \tag{E29}$$

$$P_{14}^{k+1,k} = P_{14}^{k,k} + T\, P_{34}^{k,k} \tag{E30}$$

$$P_{22}^{k+1,k} = P_{22}^{k,k} + 2T\, P_{24}^{k,k} + T^2 P_{44}^{k,k} + (T^4/4)V_{2k} \qquad \text{(E31)}$$

$$P_{23}^{k+1,k} = P_{23}^{k,k} + T\, P_{34}^{k,k} \qquad \text{(E32)}$$

$$P_{24}^{k+1,k} = P_{24}^{k,k} + T \cdot P_{44}^{k,k} + (T^3/2)\cdot V_{2k} \qquad \text{(E33)}$$

$$P_{33}^{k+1,k} = P_{33}^{k,k} + T^2 V_{1k} \qquad \text{(E34)}$$

$$P_{34}^{k+1,k} = P_{34}^{k,k} \qquad \text{(E35)}$$

$$P_{44}^{k+1,k} = P_{44}^{k,k} + T^2 V_{2k} \qquad \text{(E36)}$$

As we can see, the time updating equations increases the covariance (except for P_{34}), while the observation integration equations reduces the covariance again. It is clear from equations (E14) through (E23) that an increasing R reduces the size of the reduction, resulting in higher values on the covariance matrix elements. The exact dependence on $P_{k,k}$ of R, however, has to be decided through simulations.

APPENDIX F

CALCULATION OF INITIALIZATION VALUES FOR RANGE AND VELOCITY. NUMERICAL EXAMPLE.

In order to demonstrate the use of the results from section 4.3 and 4.4, the following example is constructed:

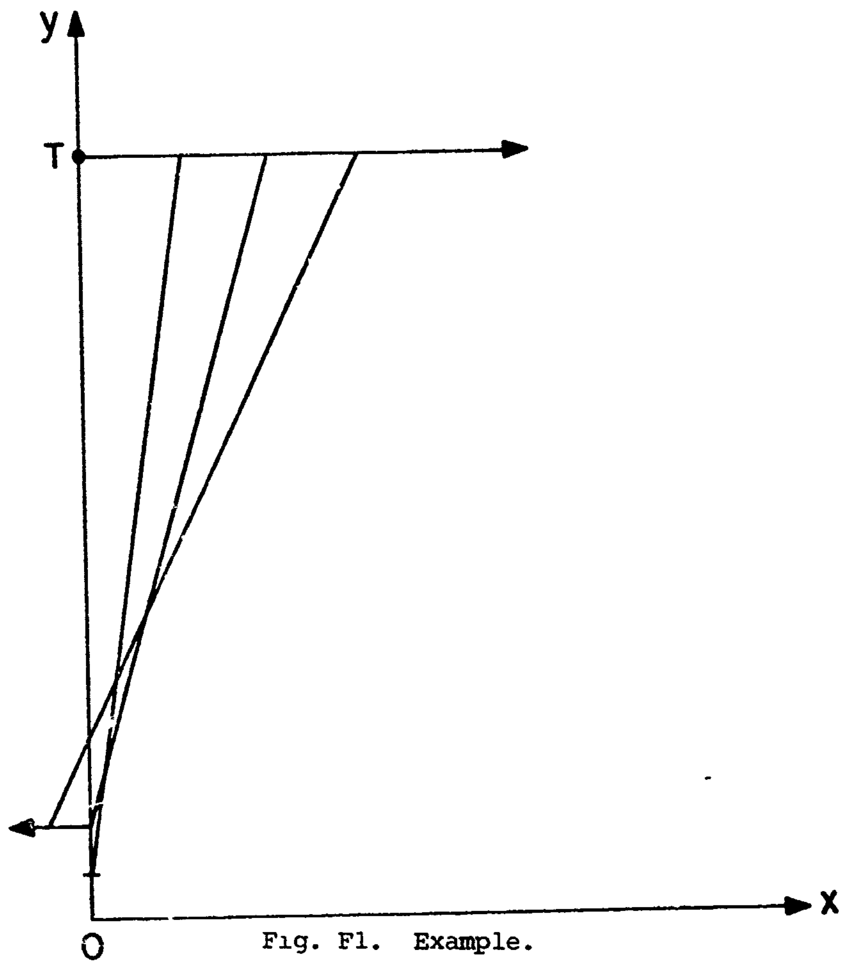


Fig. F1. Example.

In Fig. F1 we have:

$R_0 = 5000\text{m}$ $\qquad v_{sx_0} = v_{sx_1} = 0$ m/sec

$v_x = 20$ m/sec $\qquad v_{sy_0} = v_{sy_1} = 10$ m/sec

$v_y = 0$ $\qquad v_{sx_2} = -10$ m/sec

$\Delta t = 25$ sec $\qquad v_{sy_2} = 0$ m/sec

Based on this data, we can calculate Δx_{si}, Δy_{si}, $i = 0,1,2$, and ϕ_i, $i = 0,1,2,3$. We have:

$$\Delta x_{s0} = \Delta x_{s1} = \Delta y_{s2} = 0$$

$$\Delta y_{s0} = \Delta y_{s1} = 250 \text{ m}$$

$$\Delta x_{s2} = -250 \text{ m}$$

$$\phi_0 = 0°$$

$$\phi_1 = \tan^{-1}\left[\frac{500}{5000-250}\right] \approx 6.01°$$

$$\phi_2 = \tan^{-1}\left[\frac{500}{5000-500}\right] \approx 12.53°$$

$$\phi_3 = \tan^{-1}\left[\frac{1750}{500-500}\right] \approx 21/25°$$

Equation (4.51) then gives

$$R_0 = \frac{500 \sin(\phi_2-\phi_1)[\sin\phi_3-\cos\phi_3]}{\sin(\phi_3-\phi_0)\sin(\phi_2-\phi_1)-3\sin(\phi_1-\phi_0)\sin(\phi_3-\phi_2)} \cong 5001.54 \text{ m}$$

From equation (4.25) and (4.24) we obtain:

$$v_x = \frac{1}{50} \cdot R_0\left(\frac{\sin(\phi_1-\phi_0)\sin\phi_2}{\sin(\phi_2-\phi_1)} - \sin\phi_0\right) \approx 20.01 \text{ m/sec}$$

$$v_y = \frac{1}{50}\left[R_0\left(\frac{\sin(\phi_1-\phi_0)\ \phi}{\sin(\phi_2-\phi_1)} - \cos\ \phi_0\right) + 500\right] \approx 0.01\ \text{m/sec}$$

APPENDIX G

CALCULATION OF INITIAL VARIANCE FOR RANGE, $\sigma^2_{R_0}$

If range, R_0, is calculated from equation (4.49), the initial value of $\sigma^2_{R_0}$ can be calculated.

We assume that the different bearing observations are statistical independent and uncorrelated, with variance σ^2_{ϕ}. The bearing observations are further statistical independent and uncorrelated with the observers position increments, Δx_i, Δy_i, $i=0,\ldots,2$. The different position increments are also assumed statistical independent and uncorrelated, with variance $\sigma^2_{xs} = \sigma^2_{ys} = \sigma^2_{s}$.

We now define:

$$x1 = \Delta t_1 [\Delta t_3 (\Delta x_{s0} + \Delta x_{s1}) - (t_2 - t_0)\Delta x_{s2}] \tag{G1}$$

$$y1 = \Delta t_1 [\Delta t_3 (\Delta y_{s0} + \Delta y_{s1}) - (t_2 - t_0)\Delta y_{s2}]$$

$$x2 = (t_3 - t_0)[\Delta t_2 \Delta x_{s0} - \Delta t_1 \Delta x_{s1}] \tag{G3}$$

$$y2 = (t_3 - t_0)[\Delta t_2 \Delta y_{s0} - \Delta t_1 \Delta y_{s1}] \tag{G4}$$

Now, by use of equations (G1)-(G4), equation (4.49) can be written:

$$R_0 = \frac{\sin(\phi_2-\phi_1)[y1\cdot\sin\phi_3 - x1\cdot\cos\phi_3] + \sin(\phi_3-\phi_2)[x2\cos\phi_1 - y2\sin\phi_1]}{\Delta t,\Delta t_3 \sin(\phi_3-\phi_0)\sin(\phi_2-\phi_1) - \Delta t_2 (t_3-t_0)\cdot\sin(\phi_1-\phi_0)\cdot\sin(\phi_3-\phi_2)} \tag{G5}$$

As we can see, equation (G5) can be expressed as:

$$R_0 = \underline{f}(\phi_0,\phi_1,\phi_2,\phi_3,x1,x2,y1,y2) \tag{G6}$$

If we develope the Taylor expansion for R_0 about some nominal value $\overline{R}_0$, given by:

$$\overline{R}_0 = \underline{f}(\overline{\phi}_0,\ \overline{\phi}_1,\ \overline{\phi}_2,\ \phi_3,\ \overline{x1},\ \overline{x2},\ \overline{y1},\ \overline{y2}) \tag{G7}$$

and neglecting terms higher than first order, we get:

$$R_0 = \overline{R}_0 + \frac{\partial f}{\partial\phi_0}\,\delta\phi_0 + \frac{\partial f}{\partial\phi_1}\,\delta\phi_1 + \frac{\partial f}{\partial\phi_2}\,\partial\phi_3 + \frac{\partial f}{\partial\phi_3}\,\delta\phi_3 + \frac{\partial f}{\partial x_1}\,\delta x_1 + \frac{\partial f}{\partial x_2}\,\delta x_2$$

$$+ \frac{\partial f}{\partial y_1}\,\delta y_1 + \frac{\partial f}{\partial y_2}\,\delta y_2 \tag{G8}$$

We now define:

$$\delta R_0 = R_0 - \overline{R}_0 \tag{G9}$$

We then have:

$$\sigma^2_{R_0} = E\{\delta R_0^2\} \tag{G10}$$

By making use of equation (G8), (G9) and (G10) we get:

$$\sigma^2_{R_0} = \left[\left(\frac{\partial f}{\partial\phi_0}\right)^2 + \left(\frac{\partial f}{\partial\phi_1}\right)^2 + \left(\frac{\partial f}{\partial\phi_2}\right)^2 + \left(\frac{\partial f}{\partial\phi_3}\right)^2\right]\cdot\sigma^2_\phi + \left(\frac{\partial f}{\partial x_1}\,\sigma_{x1}\right)^2 + \left(\frac{\partial f}{\partial y_1}\,\sigma_{y1}\right)^2 +$$

$$\left(\frac{\partial f}{\partial x_2}\,\sigma x_2\right)^2 + \left(\frac{\partial f}{\partial y_2}\,\sigma_{y2}\right)^2 \tag{G11}$$

Now, naming the demoninator in equation (G5) D, and the nominator N, we calculate the different derivatives of the f-function. We have:

$$\frac{\partial f}{\partial \phi_0} = -\frac{N}{D^2} \cdot [-\Delta t_1\, \Delta t_3\, \cos(\phi_3-\phi_0)\sin(\phi_2-\phi_1) + \Delta t_2(t_3-t_0)\cos(\phi_1-\phi_0)$$
$$\sin(\phi_3-\phi_2)] \tag{G12}$$

$$\frac{\partial f}{\partial \phi_1} = \frac{1}{D^2}\, [-(+\cos(\phi_2-\phi_1)[y1\, \sin\phi_3 - x1\, \cos\phi_3] + \sin(\phi_3-\phi_2)[x2\, \sin\phi_1 + y2\cos\phi_1]) \cdot D +$$
$$N \cdot (\Delta t_1 \Delta t_3 \sin(\phi_3-\phi_0)\cos(\phi_2-\phi_1) + \Delta t_2(t_3-t_0)\cos(\phi_1-\phi_0)\sin(\phi_3-\phi_2))] \tag{G13}$$

$$\frac{\partial f}{\partial \phi_2} = \frac{1}{D^2}[(\cos(\phi_2-\phi_1)[y1 \cdot \sin\phi_3 - x1\, \cos\phi_3] - \cos(\phi_3-\phi_2)[x2\, \cos\phi_1 - y2\, \sin\phi_1]) \cdot D -$$
$$N \cdot (\Delta t_1 \cdot \Delta t_3\, \sin(\phi_3-\phi_0)\cos(\phi_2-\phi_1) + \Delta t_2(t_3-t_0)\sin(\phi_1-\phi_0)\cos(\phi_3-\phi_2))] \tag{G14}$$

$$\frac{\partial f}{\partial \phi_3} = \frac{1}{D^2}[(\sin(\phi_2-\phi_1)[y1\, \cos\phi_3 + x1\, \sin\phi_3] + \cos(\phi_3-\phi_2)[x2\, \cos\phi_1 - y2\, \sin\phi_1]) \cdot D -$$
$$N(\Delta t_1 \Delta t_3 \cos(\phi_3-\phi_0)\sin(\phi_2-\phi_1) - \Delta t_2(t_3-t_0)\sin(\phi_1-\phi_0)\cos(\phi_3-\phi_2))] \tag{G15}$$

$$\frac{\partial f}{\partial x1} = -\frac{1}{D}\sin(\phi_2-\phi_1)\cdot\cos\,\phi_3 \tag{G16}$$

$$\frac{\partial f}{\partial y1} = \frac{1}{D}\sin(\phi_2-\phi_1)\sin\phi_3 \tag{G17}$$

$$\frac{\partial f}{\partial x2} = \frac{1}{D}\sin(\phi_3-\phi_2)\cos\phi_1 \tag{G18}$$

$$\frac{\partial f}{\partial y2} = -\frac{1}{D}\sin(\phi_3-\phi_2)\sin\phi_1 \tag{G19}$$

Further we have:

$$\sigma_{x1}^2 = \left[\left(\frac{\partial x1}{\partial \Delta x_{s0}}\right)^2 + \left(\frac{\partial x1}{\partial \Delta x_{s1}}\right)^2 + \left(\frac{\partial x1}{\partial \Delta x_{s2}}\right)^2\right]\sigma_s^2 \tag{G20a}$$

$$\sigma_{x1}^2 = \Delta t_1^2 \cdot [\Delta t_3^2 + \Delta t_3^2 + (t_2-t_0)^2]\sigma_s^2 \tag{G20b}$$

$$\sigma_{y1}^2 = \left[\left(\frac{\partial y1}{\partial \Delta y_{s0}}\right)^2 + \left(\frac{\partial y1}{\partial \Delta y_{s1}}\right)^2 + \left(\frac{\partial y1}{\partial \Delta x_{s2}}\right)^2\right]\sigma_s^2 \tag{G21a}$$

$$\sigma_{y1}^2 = \Delta t_1^2[\Delta t_3^2 + \Delta t_3^2 + (t_2-t_0)^2]\sigma_s^2 \tag{G21b}$$

$$\sigma_{x2}^2 = \left[\left(\frac{\partial x2}{\partial \Delta x_{s0}}\right)^2 + \left(\frac{\partial x2}{\partial \Delta x_{s1}}\right)^2\right] \cdot \sigma_s^2 = (t_3-t_0)^2(\Delta t_2^2+\Delta t_1^2)\sigma_s^2 \tag{G22}$$

$$\sigma_{y2}^2 = \left[\left(\frac{\partial y2}{\partial \Delta y_{s0}}\right)^2 + \left(\frac{\partial y2}{\partial \Delta y_{s1}}\right)^2\right] \cdot \sigma_s^2 = (t_3-t_0)^2(\Delta t_2^2+\Delta t_1^2)\sigma_s^2 \tag{G23}$$

We now define:

$$K1 = \sum_{i=0}^{3}\left(\frac{\partial f}{\partial \phi_i}\right)^2 \tag{G24}$$

and

$$K_2 \cdot \sigma_s^2 = \left(\frac{\partial f}{\partial x_1}\sigma_{x1}\right)^2 + \left(\frac{\partial f}{\partial x_2}\sigma_{x2}\right)^2 + \left(\frac{\partial f}{\partial y_1}\sigma_{y_1}\right)^2 + \left(\frac{\partial f}{\partial y_2}\sigma_{y_2}\right)^2 \tag{G25}$$

Then equation (G11) can be written

$$\sigma^2_{R_0} = K1\ \sigma^2_\phi + K2\ \sigma^2_s \tag{G26}$$

The variable K2 can be calculated to be:

$$K_2 = \frac{1}{D^2}[\sin^2(\phi_2-\phi_1)(2\Delta t_3^2 + (t_2-t_0)^2 + \sin^2(\phi_3-\phi_1)(t_3-t_0)^2(\Delta t_2^2+\Delta t_1^2)] \tag{G27}$$

Further, K1 can be expressed as:

$$K1 = \frac{1}{D^4}[N^2 K3 - 2N\cdot D\cdot K4 + D^2 K5] \tag{G28}$$

where K3, K4 and K5 are given by:

$$\begin{aligned} K3 = 2\{&\Delta t_1^2\Delta t_3^2[\cos^2(\phi_3-\phi_0)\sin^2(\phi_2-\phi_1) + \sin^2(\phi_3-\phi_0)\cos^2(\phi_2-\phi_1)] \\ &+ \Delta t_1\Delta t_2\Delta t_3(t_3-t_0)\sin^2(\phi_1+\phi_3-\phi_0-\phi_2) \\ &+ \Delta t_2^2(t_3-t_0)^2[\cos^2(\phi_1-\phi_0)\sin^2(\phi_3-\phi_2) + \sin^2(\phi_1-\phi_0)\cos^2(\phi_3-\phi_2)]\} \end{aligned} \tag{G29}$$

$$\begin{aligned} K4 = \Delta t_1\Delta t_3\{&2(y1\ \sin\phi_3-x1\ \cos\phi_3)\sin(\phi_3-\phi_0)\ \cos^2(\phi_2-\phi_1) + \\ &[y1\ \cos\phi_3 + x1\ \sin\phi_3]\ \cos(\phi_3-\phi_0)\sin^2(\phi_2-\phi_1) - [x2\ \cos\phi_1-y2\ \sin\phi_1]. \\ &\cos(\phi_3-\phi_2)\sin(\phi_1+\phi_3-\phi_0-\phi_2) + [x2\ \sin\phi_1+y2\ \cos\phi_1]\sin(\phi_3-\phi_2)\cos(\phi_2-\phi_1) \\ &\sin(\phi_3-\phi_0)\} \\ &+ \Delta t_2(t_3-t_0)\{[y1\ \sin\phi-x1\ \cos\phi_3]\cos(\phi_2-\phi_1)\sin(\phi_1+\phi_3-\phi_0-\phi_2) \\ &-[y1\ \cos\phi_3 + x1\ \sin\phi_3]\ \sin(\phi_2-\phi_1)\sin(\phi_1-\phi_0)\cos(\phi_3-\phi_2)\} \end{aligned} \tag{G30}$$

$$K5 = \sin^2(\phi_2-\phi_1)[y1\ \cos\phi_3 + x1\ \sin\phi_3]^2 + \sin^2(\phi_3-\phi_2)[x2\ \sin\phi_1+y2\ \cos\phi_1]^2$$

$$+ 2\{\cos^2(\phi_2-\phi_1)[y1\ \sin\phi_3-x1\ \cos\phi_3]^2+\cos^2(\phi_3-\phi_2)[x2\ \cos\phi_1-y2\ \sin\phi_1]^2$$

$$-\cos(\phi_2-\phi_1)\sin(\phi_3-\phi_2)[y1\ \sin\phi_3-X1\ \cos\phi_3][x2\ \sin\phi_1+y2\ \cos\phi_1]$$

$$-\cos(\phi_2-\phi_1)\cos(\phi_3-\phi_2)[y1\ \sin\phi_3-X1\ \cos\phi_3][x2\ \cos\phi_1-y2\ \sin\phi_1]$$

$$+ \sin(\phi_2-\phi_1)\cos(\phi_3-\phi_2)[y1\ \cos\phi_3+x1\ \sin\phi_3][x2\ \cos\phi_1-y2\ \sin\phi_1]\}$$

(G31)

APPENDIX H

INITIAL VALUES FOR THE VELOCITY COMPONENTS OF THE COVARIANCE MATRIX

From section 4.3, equations (4.24) and (4.25), we have the following expressions for the target velocity components:

$$\begin{aligned} v_x = \frac{1}{t_2-t_0} \Bigg[& R_0 \left\{ \frac{\Delta t_2}{\Delta t_1} \; \frac{\sin(\phi_1-\phi_0)\sin\phi_2}{\sin(\phi_2-\phi_1)} - \sin \phi_0 \right\} + \Delta x_{s0} + \Delta x_{s1} + \\ & \frac{\sin\phi_2}{\sin(\phi_2-\phi_1)} \left\{ \cos \phi_1 \left[\frac{\Delta t_2}{\Delta t_1} \Delta x_{s0} - \Delta x_{s1} \right] - \sin\phi_1 \left[\frac{\Delta t_2}{\Delta t_1} \Delta y_{s0} - \Delta y_{s1} \right] \right\} \Bigg] \end{aligned} \tag{H1}$$

$$\begin{aligned} v_y = \frac{1}{t_2-t_0} \Bigg[& R_0 \left\{ \frac{\Delta t_2}{\Delta t_1} \; \frac{\sin(\phi_1-\phi_0)\cos\phi_2}{\sin(\phi_2-\phi_1)} - \cos\phi_0 \right\} + \Delta y_{s0} + \Delta y_{s1} + \\ & \frac{\cos \phi_2}{\sin(\phi_2-\phi_1)} \left\{ \cos \phi_1 \left[\frac{\Delta t_2}{\Delta t_1} \Delta x_{s0} - \Delta x_{s1} \right] - \sin\phi_1 \left[\frac{\Delta t_2}{\Delta t_1} \Delta y_{s0} - y_{s1} \right] \right\} \Bigg] \end{aligned} \tag{H2}$$

The following accuracies are assumed defined:

$$\left. \begin{aligned} & \sigma_{sx} = \sigma_{sy} = \sigma_s \\ & \sigma_{R_0} \\ & \sigma_{\phi_0} = \sigma_{\phi_1} = \sigma_{\phi_2} = \sigma_\phi \end{aligned} \right\} \tag{H3}$$

Now, in order to simplify the equatiions to be developed, equations (H1) and (H2) can be written:

$$v_x = f_1(\Delta x_{si}, \Delta y_{si}, R_0, \phi_0, \phi_1, \phi_2) \tag{H4}$$

$$v_y = f_2(\Delta x_{si}, \Delta y_{si}, R_0, \phi_0, \phi_1, \phi_2) \tag{H5}$$

where i=0,1.

We now develope the first order Taylor expansion for f_1 and f_2 about some nominal values for Δx_{s0}, Δx_{s1}, Δy_{s0}, Δy_{s1}, R_0, ϕ_0, ϕ_1, ϕ_2, named $\Delta\overline{x}_{s0}$, $\Delta\overline{x}_{s1}$, $\Delta\overline{y}_{s0}$, $\Delta\overline{y}_{s1}$, $\overline{R}_0$, $\overline{\phi}_0$, $\overline{\phi}_1$ and $\overline{\phi}_2$. Then we will get the following equations:

$$\delta v_x = \frac{\partial f_1}{\partial \Delta x_{s0}} \varepsilon\Delta x_{s0} + \frac{\partial f_1}{\partial \Delta x_{s1}} \delta\Delta x_{s1} + \frac{\partial f_1}{\partial \Delta y_{s0}} \delta\Delta y_{s0} + \frac{\partial f_1}{\partial \Delta y_{s1}} \delta\Delta y_{s1} + \frac{\partial f_1}{\partial R_0} \delta R_0$$
$$+ \frac{\partial f_1}{\partial \phi_0} \delta\phi_0 + \frac{\partial f_1}{\partial \phi_1} \delta\phi_1 + \frac{\partial f_1}{\partial \phi_2} \delta\phi_2 \tag{H6}$$

$$\delta v_y = \frac{\partial f_2}{\partial \Delta x_{s0}} \delta\Delta x_{s0} + \frac{\partial f_2}{\partial \Delta x_{s1}} \delta\Delta x_{s1} + \frac{\partial f_2}{\partial \Delta y_{s0}} \delta\Delta y_{s0} + \frac{\partial f_2}{\partial \Delta y_{s1}} \delta\Delta y_{s1} + \frac{\partial f_2}{\partial R_0} \delta R_0$$
$$+ \frac{\partial f_2}{\partial \phi_0} \delta\phi_0 + \frac{\partial f_2}{\partial \phi_1} \delta\phi_1 + \frac{\partial f_2}{\partial \phi_2} \delta\phi_2 \tag{H7}$$

where:

$$\delta v_x = v_x - \overline{v}_x = v_x - f_1(\overline{\cdot}) \tag{H8}$$

$$\delta v_y = v_y - \overline{v}_y = v_y - f_2(\overline{\cdot}) \tag{H9}$$

$$\delta\Delta x_{s0} = \Delta x_{s0} - \Delta x_{s0} \qquad \text{(H10)}$$

$$\delta\Delta x_{s1} = \Delta x_{s1} - \Delta\overline{x}_{s1}$$

$$\delta\Delta y_{s0} = \Delta y_{s0} - \Delta\overline{y}_{s0} \qquad \text{(H12)}$$

$$\delta\Delta y_{s1} = \Delta y_{s1} - \Delta\overline{y}_{s1} \qquad \text{(H13)}$$

$$\delta R_0 = R_0 - \overline{R}_0 \qquad \text{(H14)}$$

$$\delta\phi_0 = \phi_0 - \overline{\phi}_0 \qquad \text{(H15)}$$

$$\delta\phi_1 = \phi_1 - \overline{\phi}_1 \qquad \text{(H16)}$$

$$\delta\phi_2 = \phi_2 - \overline{\phi}_2 \qquad \text{(H17)}$$

The velocity components of the covariance matrix are now given by:

$$\begin{bmatrix} p_{33} & p_{34} \\ p_{43} & p_{44} \end{bmatrix} = \begin{bmatrix} E\{(\delta v_x)^2\} & E\{\delta v_x \delta v_y\} \\ E\{\delta v_y \delta v_x\} & E\{(\delta v_y)^2\} \end{bmatrix} \qquad \text{(H18)}$$

We now assume that Δx_{s0}, Δx_{s1}, Δy_{s0}, Δy_{s1}, R_0, ϕ_0, ϕ_1 and ϕ_2 all are statistical independent. Then we have

$$p_{33} = E\{(\delta v_x)^2\} = \left[\left(\frac{\partial f_1}{\partial\Delta x_{s0}}\right)^2 + \left(\frac{\partial f_1}{\partial\Delta x_{s1}}\right)^2 + \left(\frac{\partial f_1}{\partial\Delta y_{s0}}\right)^2 + \left(\frac{\partial f_1}{\partial\Delta y_{s1}}\right)^2\right]\cdot \sigma_s^2 +$$

$$\left(\frac{\partial f_1}{\partial R_0}\right)^2 \cdot \sigma_{R_0}^2 + \left[\left(\frac{\partial f_1}{\partial\phi_0}\right)^2 + \left(\frac{\partial f_1}{\partial\phi_1}\right)^2 + \left(\frac{\partial f_1}{\partial\phi_2}\right)^2\right]\sigma_\phi^2 \qquad \text{(H19)}$$

$$P_{44} = E\{(\delta v_y)^2\} = \left[\left(\frac{\partial f_2}{\partial \Delta x_{s0}}\right)^2 + \left(\frac{\partial f_2}{\partial \Delta x_{s1}}\right)^2 + \left(\frac{\partial f_2}{\partial \Delta y_{s0}}\right)^2 + \left(\frac{\partial f_2}{\partial \Delta y_{s1}}\right)^2\right]\sigma_s^2 +$$

$$\left(\frac{\partial f_2}{\partial R_0}\right)^2 \cdot \sigma_{R_0}^2 + \left[\left(\frac{\partial f_2}{\partial \phi_0}\right)^2 + \left(\frac{\partial f_2}{\partial \phi_1}\right)^2 + \left(\frac{\partial f_2}{\partial \phi_2}\right)^2\right]\sigma_\phi^2 \qquad \text{(H20)}$$

$$P_{34} = P_{43} = E\{\delta v_x \delta v_y\} = \left[\frac{\partial f_1}{\partial \Delta x_{s0}} \cdot \frac{\partial f_0}{\partial \Delta x_{s0}} + \frac{\partial f_1}{\partial \Delta x_{s1}} \cdot \frac{\partial f_2}{\partial \Delta x_{s1}} \cdot\right.$$

$$\left.\frac{\partial f_1}{\partial \Delta y_{s0}} \cdot \frac{\partial f_2}{\partial \Delta y_{s0}} + \frac{\partial f_1}{\partial \Delta y_{s1}} \cdot \frac{\partial f_2}{\partial \Delta y_{s1}}\right]\sigma_s^2 +$$

$$\frac{\partial f_1}{\partial R_0} \cdot \frac{\partial f_2}{\partial R_0}\sigma_{R_0}^2 + \left[\frac{\partial f_1}{\partial \phi_0}\frac{\partial f_2}{\partial \phi_0} + \frac{\partial f_1}{\partial \phi_1}\frac{\partial f_2}{\partial \phi_1} + \frac{\partial f_1}{\partial \phi_2}\frac{\partial f_2}{\partial \phi_2}\right]\sigma_\phi^2 \qquad \text{(H21)}$$

The different derivatives of f_1 and f_2 are given in the following:

$$\frac{\partial f_1}{\partial \Delta x_{s0}} = \frac{1}{t_2 - t_0}\left[1 + \frac{\sin\phi_2 \cos\phi_1}{\sin(\phi_2-\phi_1)} \cdot \frac{\Delta t_2}{\Delta t_1}\right] \qquad \text{(H22)}$$

$$\frac{\partial f_1}{\partial \Delta x_{s1}} = \frac{1}{t_2 - t_0}\left[1 - \frac{\sin\phi_2 \cos\phi_1}{\sin(\phi_2-\phi_1)}\right] \qquad \text{(H23)}$$

$$\frac{\partial f_1}{\partial \Delta y_{s0}} = -\frac{1}{t_2 - t_0} \cdot \frac{\sin\phi_2}{\sin(\phi_2-\phi_1)} \cdot \sin\phi_1 \frac{\Delta t_2}{\Delta t_1} \qquad \text{(H24)}$$

$$\frac{\partial f_1}{\partial \Delta y_{s1}} = \frac{1}{t_2 - t_0} \frac{\sin\phi_2}{\sin(\phi_2-\phi_1)} \cdot \sin\phi_1 \qquad \text{(H25)}$$

$$\frac{\partial f_2}{\partial \Delta x_{s0}} = \frac{1}{t_2 - t_0} \cdot \frac{\cos\phi_2 \cos\phi_1}{\sin(\phi_2-\phi_1)} \cdot \frac{\Delta t_2}{\Delta t_1} \qquad \text{(H26)}$$

$$\frac{\partial f_2}{\partial \Delta x_{s1}} = - \frac{1}{t_2-t_0} \cdot \frac{\cos \phi_2 \cos \phi_1}{\sin(\phi_2-\phi_1)} \tag{H27}$$

$$\frac{\partial f_2}{\partial \Delta y_{s0}} = \frac{1}{t_2-t_0} \left[1 - \frac{\cos \phi_2 \cdot \sin \phi_1}{\sin(\phi_2-\phi_1)} \cdot \frac{\Delta t_2}{\Delta t_1}\right] \tag{H28}$$

$$\frac{\partial f_2}{\partial \Delta y_{s1}} = \frac{1}{t_2-t_0} \left[1 + \frac{\cos \phi_2 \sin \phi_1}{\sin(\phi_2-\phi_1)}\right] \tag{H29}$$

$$\frac{\partial f_1}{\partial R_0} = \frac{1}{t_2-t_0} \left[\frac{\Delta t_2}{\Delta t_1} \; \frac{\sin(\phi_1-\phi_0)\sin \phi_2}{\sin(\phi_2-\phi_1)} - \sin \phi_0\right] \tag{H30}$$

$$\frac{\partial f_2}{\partial R_0} = \frac{1}{t_2-t_0} \left[\frac{\Delta t_2}{\Delta t_1} \; \frac{\sin(\phi_1-\phi_0)\cos \phi_2}{\sin(\phi_2-\phi_1)} - \cos \phi_0\right] \tag{H31}$$

$$\frac{\partial f_1}{\partial \phi_0} = \frac{R_0}{t_2-t_0} \cdot \left[\frac{\Delta t_2}{\Delta t_1} \; \frac{\cos(\phi_1-\phi_0) \sin \phi_2}{\sin(\phi_2-\phi_1)} + \cos \phi_0\right] \tag{H32}$$

$$\frac{\partial f_2}{\partial \phi_0} = \frac{R_0}{t_2-t_0} \left[\frac{\Delta t_2}{\Delta t_1} \; \frac{\cos(\phi_1-\phi_0) \cos \phi_2}{\sin(\phi_2-\phi_1)} - \sin \phi_0\right] \tag{H33}$$

$$\begin{aligned}\frac{\partial f_1}{\partial \phi_1} = \frac{1}{t_2-t_0} \cdot \Bigg[R_0 \cdot \frac{\Delta t_2}{\Delta t_1} \cdot \frac{\sin \phi_2 \cdot \sin(\phi_2-\phi_0)}{\sin^2(\phi_2-\phi_1)} - \frac{\sin \phi_2}{\sin^2(\phi_2-\phi_1)} \cdot \Bigg\{ \\ -\left[\frac{\Delta t_2}{\Delta t_1} \Delta x_{s0} - \Delta x_{s1}\right]\cos \phi_2 + \left[\frac{\Delta t_2}{\Delta t_1} \Delta y_{s0} - \Delta y_{s1}\right] \sin \phi_2\Bigg\}\Bigg]\end{aligned} \tag{H34}$$

$$\begin{aligned}\frac{\partial f_2}{\partial \phi_1} = \frac{1}{t_2-t_0} \Bigg[R_0 \frac{\Delta t_2}{\Delta t_1} \cdot \frac{\cos \phi_2 \cdot \sin(\phi_2-\phi_0)}{\sin^2(\phi_2-\phi_1)} + \frac{\cos \phi_2}{\sin^2(\phi_3-\phi_1)} \cdot \Bigg\{ \\ -\left[\frac{\Delta t_2}{\Delta t_1} \Delta x_{s0} - \Delta x_{s1}\right]\cos \phi_2 + \left[\frac{\Delta t_2}{\Delta t_1} \Delta y_{s0} - \Delta y_{s1}\right]\sin \phi_2\Bigg\}\Bigg]\end{aligned} \tag{H35}$$

$$\frac{\partial f_1}{\partial \phi_2} = -\frac{1}{t_2-t_0}\left[R_0 \cdot \frac{\Delta t_2}{\Delta t_1} \frac{\sin(\phi_1-\phi_0)\sin\phi_1}{\sin^2(\phi_2-\phi_1)} + \frac{\sin\phi_1}{\sin^2(\phi_2-\phi_1)} \cdot \left\{ \cos\phi_1\left[\frac{\Delta t_2}{\Delta t_1} \Delta x_{s0} - \Delta x_{s1}\right] - \sin\phi_1 \left[\frac{\Delta t_2}{\Delta t_1} \Delta y_{s0} - \Delta y_{s1}\right]\right\}\right] \tag{H36}$$

$$\frac{\partial f_2}{\partial \phi_2} = -\frac{1}{t_2-t_1} \cdot \left[R_0 \cdot \frac{\Delta t_2}{\Delta t_1} \frac{\sin(\phi_1-\phi_0)\cos\phi_1}{\sin^2(\phi_2-\phi_1)} + \frac{\cos\phi}{\sin^2(\phi_2-\phi_1)} \cdot \left\{ \cos\phi_1\left[\frac{\Delta t_2}{\Delta t_1} \Delta x_{s0} - \Delta x_{s1}\right] - \sin\phi_1\left[\frac{\Delta t_2}{\Delta t_1} \Delta y_{s0} - \Delta y_{s1}\right]\right\}\right] \tag{H37}$$